

METHOD AND APPARATUS FOR SPAN AND SUBSPAN SORTING RENDERING SYSTEM

Publication number: JP11515121T

Publication date: 1999-12-21

Inventor:

Applicant:

Classification:

- international: G09G5/36; G06T11/40; G06T15/00; G09G5/36; G06T11/40; G06T15/00; (IPC1-7): G06T11/40; G06T15/00; G09G5/36

- European: G06T15/00A

Application number: JP19960507824T 19960726

Priority number(s): WO1996US12390 19960726; US19950001529P 19950726

Also published as:



WO9705576 (A1-corr)

WO9705576 (A1)

WO9705575 (A1)

EP0870282 (A1-corr)

EP0870282 (A1)

[more >>](#)

[Report a data error here](#)

Abstract not available for JP11515121T

Abstract of corresponding document: **WO9705576**

The invention provides for a data shifting capability that permits sorting the data in addition to searching for obtaining real-time performance in color, with high quality imagery through a simple search of a spatial database based on a rectangularly shaped search region or range search (fig. 13). A sorting Magnitude Comparison Content Addressable Memory (SMCCAM) performs a range search, introducing a conservative approximation of the ideal Occluding Region, and provides a MCCAM wherein the data words stored in the fields is shifted to corresponding fields in an adjacent word, based on the magnitude comparisons (fig. 7). The 3D graphics method stores the parameters of a polygon span in a spatial database (804-814) and a query operation is performed on the database to determine which of those spans, or portions of spans, are visible (816 and 818), and applies a rule for comparing new span portion to an old span portion on a subspan-by-subspan basis, thereby providing additional polygon edge information within a raster line, providing anti-aliasing.

Data supplied from the **esp@cenet** database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表平11-515121

(43) 公表日 平成11年(1999)12月21日

(51) Int.Cl. ⁶	識別記号	F I	
G 0 6 T 11/40		G 0 6 F 15/72	4 0 0
15/00		G 0 9 G 5/36	5 1 0 V
G 0 9 G 5/36	5 1 0	G 0 6 F 15/72	4 5 0 A

審査請求 未請求 予備審査請求 有 (全 199 頁)

(21) 出願番号 特願平9-507824
 (86) (22) 出願日 平成8年(1996)7月26日
 (85) 翻訳文提出日 平成10年(1998)1月26日
 (86) 国際出願番号 PCT/US96/12390
 (87) 国際公開番号 WO97/05576
 (87) 国際公開日 平成9年(1997)2月13日
 (31) 優先権主張番号 60/001, 529
 (32) 優先日 1995年7月26日
 (33) 優先権主張国 米国 (US)

(71) 出願人 レイカー, インコーポレイティド
 アメリカ合衆国, カリフォルニア 94304,
 バロアルト, ノース カリフォルニア ア
 ベニュー 950
 (72) 発明者 ダラク, ジェローム, エフ., ジュニア
 アメリカ合衆国, カリフォルニア 94304,
 バロアルト, ノース カリフォルニア ア
 ベニュー 950
 (74) 代理人 弁理士 石田 敬 (外3名)

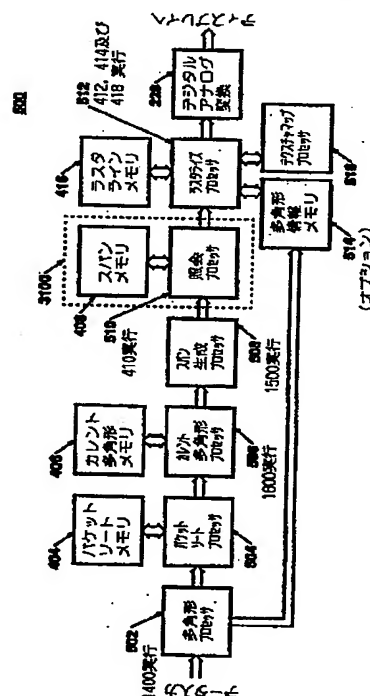
最終頁に続く

(54) 【発明の名称】 スパンおよびサブスパン・ソーティング・レンダリング・システムのための方法および装置

(57) 【要約】

本発明は、矩形形状検索領域に基づく空間データベースの単純検索または範囲検索によって、高品質画像のカラーでのリアルタイムパフォーマンスを得るため、検索に加えてデータをソートできるデータシフティングケイバビリティを提供する (図13)。ソーティング・マグニチュード連想メモリ (SMCCAM) は範囲検索を行って理想的な閉塞領域の保守的近似を導入し、MCCAMを提供するが、ここでフィールドに保存されたデータワードは、マグニチュード比較に基づき、隣接するワードの対応するフィールドにシフトされる (図7)。この3Dグラフィックス方法は空間データベースに多角形スパンのパラメータを保存し (804-814)、データベースに照会オペレーションを実行して、これらスパン、またはスパンポーションのどれが可視であるか決定し (816および818)、サブスパン毎に新しいスパンポーションを古いスパンポーションと比較するルールを適用し、ラスタライン内の追加多角形辺情報を与え、アンチエイリアシングを行う。

Figure 5



23. 前記多角形パラメータが、オブジェクト精密スクリーン座標での前記多角形頂点の位置、カラー情報、各多角形辺の辺導関数、開始ラスタライン、終了ラスタライン、およびスパン導関数を含む、請求の範囲第14項に記載の方法。

24. 前記スパンメモリが、第1および第2バンクを持つダブルバッファされたスパンメモリからなり、前記スパン幾何学的特性が前記第1バンクに書き込まれ、前記第1バンクへのスパン幾何学的特性の書き込みと同時に、前記第2バンクのスパン幾何学的特性が、可視スパンポーションを識別するために処理される、請求の範囲第14項に記載の方法。

25. 前記レンダリングは、リアルタイムの画像生成を提供するのに十分な速度で実行される、請求の範囲第14項に記載の方法。

26. 前記新しいソートリストは、保存された1データ値あたり約2クロックサイクル内で生成され、第1の前記クロックサイクル中、どの前記ワードが前記条件付きシフトを実行するかを照会オペレーションが決定し、第2の前記クロックサイクル中、前記スパンの前記幾何学的特性が前記スパンメモリに書き込まれる、請求の範囲第14項に記載の方法。

27. 隠面消去がピクセルではなくスパンの処理で実行されることにより、深さ複雑性の大きいシーンの計算が減らされる、請求の範囲第14項に記載の方法。

28. 前記多角形スパンはSMCCAMに保存され、照会を実行する前記ステップは、前記SMCCAMに保存された多角形スパンセット全体を直接照会して可視スパンを識別し、比較オペレーション数を、スパン総数より小さい可視多角形スパン数に比例させるステップをさらに具備する、請求の範囲第14項に記載の方法。

【発明の詳細な説明】

スパンおよびサブスパン・ソーティング・レンダリング・システムのための方法
および装置

発明の分野

本発明の分野は、1) 3次元コンピュータグラフィックス、および2) コンピュータメモリの2部分、さらに詳しく言うと連想メモリ (CAM) に関する。

発明の背景

3次元コンピュータグラフィックス

コンピュータグラフィックスは、コンピュータを使って図を生成する芸術および科学である。図または画像の生成はしばしばレンダリングと呼ばれる。一般に、3次元 (3D) コンピュータグラフィックスでは、あるシーンのオブジェクトの表面 (またはボリューム) を表す幾何形状を、フレームバッファに保存したピクセルに変換してから、ディスプレイ装置で表示する。コンピュータモニタとして用いられるCRTなどのリアルタイムディスプレイ装置が、画像を繰り返し連続的に表示することによって表示をリフレッシュする。このリフレッシュは通常、列毎に発生し、各列をラスタラインまたはスキャンラインと呼ぶ。本書では、ラスタラインは下から上に番号付けられているが、上から下の順に表示される。

3Dアニメーションでは、画像シーケンスが表示され、3次元空間でモーションの錯覚を与える。インタラクティブな3Dコンピュータグラフィックスでは、ユーザはリアルタイムで自分のビューイングポイントを変えたり幾何形状を変えたりすることができるため

、レンダリングシステムはリアルタイムにオンザフライ式に新しい画像を作成する必要がある。

3Dコンピュータグラフィックスでは、レンダリング可能なオブジェクトは一般にそれぞれ独自の局所的オブジェクト座標系を持っているため、オブジェクト座標からピクセル表示座標へ翻訳 (または変換) する必要がある。概念的には、これは次の4ステップのプロセスである。1) オブジェクト座標から、シーン全体の座標系であるワールド座標への翻訳 (サイズ拡大または縮小のための

スケーリング含む)、2) シーンのビューイングポイントに基づくワールド座標から視点座標への翻訳、3) 視点座標から透視翻訳視点座標への翻訳で、ここでは透視スケーリング(遠いオブジェクトほど小さく見える)が実行される、4) 透視翻訳視点座標から、スクリーン座標とも呼ばれるピクセル座標への翻訳。スクリーン座標とは3次元空間中のポイントで、後で説明するように、スクリーン精密(すなわちピクセル)か、オブジェクト精密(通常は浮動小数点の高精密数)のいずれかが可能である。これら翻訳ステップは、何らかの翻訳が発生する前に適切な翻訳マトリクスを予め計算しておくことで、1または2ステップに圧縮することができる。幾何形状がスクリーン座標に入ると、ピクセルカラー値セットに分け(すなわち「ラスター化」)、フレームバッファに保存される。ピクセルカラー値の生成には、Gouraud シェーディング、Phong シェーディング、テクスチャマッピングなど様々な技術が用いられる。

先行技術のレンダリングプロセスの概要は、Wattによる「Fundamentals of Three-dimensional Computer Graphics」第5章: The Rendering Process、97-113頁、Massachusetts州Reading、Addison-Wesley Publishing Company による出版、1989年、再版1991年、ISBN0-201-15442-0(以下、Wa

tt参考文献と称する)に見られる。

図1に、固有の座標軸(x_{obj} 、 y_{obj} 、 z_{obj})を持つ3次元オブジェクトの正四面体110を示す。この3次元オブジェクト110を、(x_{eye} 、 y_{eye} 、 z_{eye})に基づくビューイングポイント130の座標系で翻訳、スケーリング、位置づけを行う。オブジェクト120をビューイングプレーン102に投影し、もって透視を補正する。この時点で、オブジェクトは2次元になったように見えるが、オブジェクトの z 座標を留保し、隠面消去技術によって後で使えるようにする。オブジェクトは最終的に(x_{screen} 、 y_{screen} 、 z_{screen})に基づくスクリーン座標に翻訳され、ここで z_{screen} がページに垂直になる。ここでオブジェクト上のポイントはピクセル位置(およびその小数部)で記述するディスプレイスクリーン104内の x 座標および y 座標と、ビューイングポイント130からの

距離のスケール化バージョンの z 座標を持つ。

1つのピクセルに幾何形状の数多くの異なる部分が影響を与えるため、シーンビューイングポイント130に最も近い表面を表す幾何形状を決定しなければならない。そのため、各ピクセルについて、ピクセル領域が定めるボリューム内の可視面がピクセルカラー値を決定する一方、隠面はピクセルに影響を与え得ない。最も近い不透明面（または、幾何形状の縁がピクセル領域を横切っている場合は面）よりビューイングポイントに近い非不透明面がピクセルカラー値に影響する一方で、他の非不透明面はすべて放棄される。本書において、「閉塞された」という用語は、他の非不透明幾何形状によって隠された幾何形状を記述するために用いられる。

可視面の決定には多くの技術が開発されており、これら技術を次の通り参照によって本書に組み入れる。Foley, van Dam, FeinerおよびHughesによる「Computer Graphics: Principles and Practice

」第15章: Visible-Surface Determination、649-720頁、第2版、Massachusetts州Reading, Addison-Wesley Publishing Companyによる出版、1990年、訂正付き再版1991年、ISBN0-201-12110-7（以下、Foley 参考文献と称する）。Foley 参考文献の650頁に、「画像精密」および「オブジェクト精密」という用語が定義され、「画像精密アルゴリズムはディスプレイ装置の解像において多く実行され、各ピクセルの可視性を決定する。オブジェクト精密アルゴリズムは各オブジェクトを定義する精密度で実行され、各オブジェクトの可視性を決定する」とされている。

レンダリング・プロセスが進むにつれ、大半の先行技術レンダラーでは、ピクセルが定めるボリュームに複数の面が交差するため、あるスクリーンピクセルのカラー値を複数回計算しなければならない。あるシーンについてピクセルをレンダリングしなければならない平均回数をシーンの深さ複雑性と呼ぶ。単純なシーンの深さ複雑性は1に近いが、複雑なシーンの深さ複雑性は10や20になることがある。シーンモデルが複雑になるにつれ、レンダラーはますます深さ複雑性の高まるシーンを処理する必要がでてくる。このように大半のレンダラーで、ある

シーンの深さ複雑性は無駄な処理の尺度となる。例えば、深さ複雑性が10のシーンでは、計算の90%が隠れたピクセルに浪費されている。この無駄な計算は単純なZバッファ技術（本書で後で説明）を用いるハードウェア・レンダラーに典型的なものだが、これはハードウェアに容易に構築できるために選ばれることが多い。Zバッファ技術より複雑な方法はこれまで一般的に複雑すぎ、コスト効果高く構築することができなかった。本発明の方法および装置の重要な特徴は、幾何形状をラスタライズする前に隠れた部分を消去することでこの無駄な計算を回避する一方

、単純なままコスト効果の高いハードウェアに構築できる点である。

表面上のポイント（しばしば多角形頂点）をスクリーン座標に翻訳する時、このポイントは次の3つの座標を持つ：1）ピクセル単位のx座標（一般に小数部を含む）、2）ピクセル単位のy座標（一般に小数部を含む）、および3）視点座標、バーチャルスクリーンからの距離、またはビューイングポイントから面の相対距離を保存するその他座標系でのポイントのz座標。本書では、正のz座標値をビューイングポイントからの「ルック方向」に用い、小さい値はビューイングポイントに位置が近いことを示す。

表面をプレーナ多角形セットで近似する時、各多角形の頂点をスクリーン座標に翻訳する。多角形の中または上のポイント（頂点以外）については、エッジウォーキング218およびスパン補間220のプロセスによって、スクリーン座標を頂点座標から補間する。このように、z座標値は一般に幾何形状をレンダリングした時各ピクセル値に（カラー値と共に）含まれる。

ジェネリック3Dグラフィックス・パイプライン

多くのハードウェアレンダラーが開発されており、参照によって次の例を本書に組み入れる：Deering およびNelsonによる「Leo: A System for Cost Effective 3D Shaded Graphics」、SIGGRAPH93議事録101-108頁、1993年8月1-6日、コンピュータ・グラフィックス議事録、年次会議シリーズ、ACM SIGGRAPH、ニューヨーク、1993年、ソフトカバー、ISBN0-201-58889-7およびCD-ROM ISBN0-201-56997-3（以下、

Deering 参考文献と称する) には、「トップレベルのほぼすべての市販3Dグラフィックス・アクセラレータがこの抽象に一致するため、真にジェネリック」とするジェネリック3Dグラ

フィックス・パイプライン200 (すなわち、レンダラーあるいはレンダリングシステム) の図が含まれ、このパイプライン図を本書に図2として再現する。このようなパイプライン図はレンダリングのプロセスを示すが、特定のハードウェアは記述しない。本書は、ジェネリック3Dグラフィックス・パイプライン200のステップを一部共有する新しいグラフィックス・パイプライン400を提示する。ジェネリック3Dグラフィックス・パイプライン200の各ステップを本書に簡単に説明し、図3のメソッドフロー図300にも示す。本書を通じて多角形の処理を想定するが、3D幾何形状を記述する他の方法も代用できる。説明を単純化するため、記述する方法では多角形の種類として三角形を用いる。

図2に示すように、ジェネリック3Dグラフィックス・パイプライン200の浮動小数点インテンシブ・ファンクション250の最初のステップは、上記に説明した変換ステップ202である。変換ステップ202は図3のメソッドフロー図300の外ループの最初のステップとしても示され、これには「次の多角形取得」も含まれる。第2のステップ、クリップテスト204は、多角形がビューボリューム106に少なくとも部分的に含まれるか否か (時には錐台の形で)、多角形をチェックする。多角形がビューボリューム106中にない場合、これは放棄され、そうでない場合は処理が継続する。第3ステップは面決定206で、ここでビューイングポイントより外を向く多角形は放棄される。一般に、面決定206はボリュームの閉じたオブジェクトにのみ適用される。第4ステップ、ライティング計算208には一般に、複数の各種光源によるGouraud シェーディングおよび/またはテクスチャマッピングのセットアップを含むが、Phong シェーディングその他の方法についてもセットアップできる。第5ステップ、クリッピング210は、ビューボリ

ーム106外の多角形部分を削除するが、これは、この部分がビューイングプレ

ーン 1 0 2 の矩形領域内に投影しないためである。一般に、多角形クリッピング 2 1 0 は、多角形をどちらもビューイングプレーン 1 0 2 の面積内に投影する 2 つの小さな多角形に分けることによって行う。多角形クリッピングは計算するには高価だが、その必要性は本発明では回避されるため、計算の節約となる。第 6 ステップの透視分割 2 1 2 では、オブジェクトのビューイングプレーン 1 0 2 への投影の透視補正を行う。この時、多角形の頂点を表すポイントを、ステップ 7 のスクリーン空間変換 2 1 4 ステップによりピクセル空間座標に変換する。第 8 ステップのインクレメンタルなレンダラのセットアップ 2 1 6 では、エッジウォーキング 2 1 8 とスパン補間 2 2 0 (例: x 、 y および z 座標; RGB カラー; テクスチャマップ空間 u および v 座標; 等) に必要な各種開始、終了および増分値を計算する。

ドローイング・インテンシブ・ファンクション 2 6 0 では、エッジウォーキング 2 1 8 は、前回生成したスパン (同じ多角形中) の値をインCREMENTすることによりディスプレイ装置の各ラスタラインの水平スパンをインクレメンタルに生成し、もって多角形の対向する辺にそって縦に「ウォーキング」する。同様に、スパン補間 2 2 0 はスパンに沿って水平に「ウォーキング」して、ビューイングポイント 1 3 0 からのピクセルの距離を示す z 座標値を初めとするピクセル値を生成する。この z 座標値を Z バッファに保存した対応する値と比較することによって、 z バッファブレンド 2 2 2 は新しいピクセル値を (そのピクセル位置で前に保存した値よりビューイングポイントに近い場合) フレームバッファ 2 2 4 に書き込むことで保存するか、あるいはその新しいピクセル値を (遠い場合) 放棄する。このステップでは、アンチエイリアシング方法 (次のセク

ションで説明) によって新しいピクセルカラーを古いピクセルカラーとブレンドすることができる。

ジェネリック 3D グラフィックス・パイプライン 2 0 0 にはダブルバッファフレームバッファ 2 2 4 が含まれるため、ダブルバッファ MUX 2 2 6 も含まれる。出力ルックアップテーブル 2 2 6 が、カラーマップ値翻訳のために含まれる。最後に、デジタルからアナログへの変換 2 2 8 によって、ディスプレイ装置に入

力するアナログ信号を生成する。

ジェネリック3Dグラフィックス・パイプライン200の大きな欠点は、そのドローイング・インテンシブ・ファンクション260が固定数の多角形についてピクセルレベルで決定的でないことである。すなわち、固定数の多角形の場合、平均多角形サイズが大きくなると、より多くのピクセルレベルの計算が必要になる。しかしながら、浮動小数点インテンシブ・ファンクション250は多角形の数に比例し、平均多角形サイズから独立している。そのため、浮動小数点インテンシブ・ファンクション250とドローイング・インテンシブ・ファンクション260の計算力のバランスが平均多角形サイズで決まることから、バランスさせるのが難しくなる。

理想的なレンダラーのピクセルドローイングの計算要件は、ビューボリューム106中のすべての多角形のピクセル総数ではなく、ディスプレイスクリーン104のピクセル数に比例することになる。この理想は、ドローイング・インテンシブ・ファンクションの大半を実行する前に隠れた幾何形状を消去するため、本発明によって達成される。本発明では、浮動小数点計算が実質的にドローイング計算量から独立しているため、計算負荷のバランスは問題とならない。

アンチエイリアシング

本書において、ピクセルとはディスプレイ装置の個々に制御可能な最小要素と定義される。しかし、画像は離散ピクセルに量子化されるため、空間的なエイリアシングが発生する。典型的なエイリアシングアーティファクトは、直線や辺がピクセルの列を斜めに横切る時に生じる「階段」効果である。理想的なアンチエイリアシング画像では、各ピクセルについてピクセル領域内の可視面による部分的カバレッジを考慮して平均カラーを計算することにより、この「階段」効果をなくしている。

レンダリング・システムによっては、ピクセルをサブピクセルに分けて各サブピクセルを独立してカラー化することによってエイリアシング効果を減少させている。この画像を表示すると、各ピクセル内の全サブピクセルがブレンドされて、そのピクセルについて平均カラーを形成する。1ピクセルあたり16のサブピ

クセルを用いるレンダラーが、Akeleyによる「RealityEngine Graphics」、SIGGRAPH93 議事録109-116頁、1993年8月1-6日、コンピュータ・グラフィックス議事録、年次会議シリーズ、ACM SIGGRAPH、ニューヨーク、1993年、ソフトカバー、ISBN0-201-58889-7およびCD-ROM ISBN0-201-5997-3（以下、Akeley参考文献と称する）に記述されている。サブピクセルを用いる欠点は、各サブピクセルでカラー値を計算することによる計算の増加である。Akeley参考文献では、計算の増加は、ピクセルが多角形の線や辺を横切った時にのみピクセルをサブピクセルに分割することで減らしている。多角形の数が増えれば、この減少も重要でなくなる。言い換えると、重なり合う多数の小さな多角形から画像が形成される場合、大半のピクセルに分割が必要となる。サブピクセルの利用は画像精密なアンチエイリアシング技術である。

先行技術の別のアンチエイリアシング方法は、アルファブレンディングの実行に用いるAバッファで（この技術はAkeley参考文献にも含まれる）、L. Carpenterによる「The A-buffer, an Antialiased Hidden Surface Method」、SIGGRAPH 1984会議議事録103-108頁（以下、Carpenter 参考文献と称する）に記述されている。Aバッファは、レンダリングした多角形によるピクセルのパーセントカバレッジを記録することでエイリアシングを減らす画像精密アンチエイリアシング技術である。この技術の主な欠点は、許容可能なアンチエイリアシング多角形を得るため、各ピクセルでフロント・ツー・バック（あるいはバック・ツー・フロント）に多角形をソートしなければならない点である。

理想的なアンチエイリアシング方法では、オブジェクト精密な計算を実行して、幾何形状の可視部分を精密に識別する。これには、多角形の辺を互いに比較して、各多角形がカバーする各ピクセルの小数部を決定することが必要となる。本発明では、各スキャンライン内でオブジェクト精密なアンチエイリアシングを実行するため、この理想を達成できる。

Zバッファ

簡単に言うと、Zバッファは、各ピクセルについてピクセルに影響する（ビュ

ーイングポイントに) 最も近い幾何形状の z 座標を保存する。そのため、新しいピクセル値が生成されると、新しい各ピクセルの z 座標を対応するZバッファでの位置と比較する。新しいピクセルの z 座標の方が小さいと(すなわちビューイングポイントに近いと)、この値をZバッファに保存し、新しいピクセルのカラー値をフレームバッファに書き込む。新しいピクセルの z 座標の方が大きいと(すなわちビューイングポイントから遠いと)、フレームバッファとZバッファの値は無変更で、新しいピクセルを放棄す

る。Zバッファは画像精密な可視面決定技術である。

先行技術Zバッファ方法を含むフロー図を図3に示す。Zバッファ隠面消去方法の大きな欠点は、隠面消去を行うには幾何形状をピクセル値に変換しなければならない点である。これは、保存/放棄決定をピクセル毎に行うためである。これとは対照的に、本発明は、ピクセルではなくスパンを処理することにより、より高いレベルで隠面消去を実行する。深さ複雑性の大きいシーンでは、ほとんどが隠れており放棄されるのにも拘わらず、ビューボリューム内の幾何形状をすべてピクセルに変換しなければならないため、ピクセル毎の隠面消去計算によって無駄な計算が増える。ハードウェアレンダリングシステムでは、ピクセルカラー生成(シェーディング、テクスチャマッピング等)はしばしばZバッファ比較テストと平行して行われるため、計算の大半がカラー生成に関連し、ピクセルの大半が放棄されるため、無駄な計算が増加する。さらに、Zバッファメモリオペレーションは読み取り-修正-書き込みサイクルであり、Zバッファメモリの入/出力バスはピクセルをフレームバッファに書き込む際、一般に2回方向を変更しなければならない、レンダラーにボトルネックを生じる。このボトルネックは、本発明の装置および方法には生じない。

先行技術のZバッファは従来のランダムアクセスメモリ(RAM)、ビデオRAM(VRAM)または専用DRAMに基づく。専用DRAMの1例が、Deering, SchlappおよびLavelleによる「FBRAM: A New Form of Memory Optimized for 3D Graphics」、SIGGRAPH94議事録167-174頁、1994年7月24-29日、コンピュータ・グラフィックス議事録、年次会議シリーズ、ACM SIGGRA

PH、ニューヨーク、1994年、ソフトカバー、ISBN0-201-60795-6に提示されている。

幾何形状データベース

レンダリング可能なシーンの生成に必要な幾何形状はデータベースに保存される。この幾何形状データベースは、グラフィックス基本立体の単純なディスプレイリストでも、階層的に編成されたデータ構造でもよい。階層的編成の幾何形状データベースでは、階層のルートはデータベース全体で、データ構造のサブノード第1層は一般に、ビューポイントから見える「ワールド」のすべてのオブジェクトである。各オブジェクトにはそれぞれサブオブジェクトが入り、サブにはサブサブオブジェクトが入っているため、オブジェクトの階層的「ツリー」となる。以下、「オブジェクト」という用語は、オブジェクトの階層ツリーにおけるノードを指す。そのため、各サブオブジェクトはオブジェクトである。「ルートオブジェクト」という用語は、データ構造のサブノードの第1層にあるノードを指す。故に、シーンの階層データベースはシーンのルートノードから始まり、オブジェクトの第1層はルートオブジェクトである。

この種の階層的データベースは、Programmer's Hierarchical Interactive System (PHIGS) および PHIGS PLUS 標準で用いられる。これら標準の説明は、T. L. J. Howard 他による「A Practical Introduction to PHIGS and PHIGS PLUS」、Addison-Wesley Publishing Company、1991年、ISBN0-201-41641-7（参照により本書に組み入れ、以下、Howard 参考文献と称する）にある。Howard 参考文献は、5頁から8頁に3Dモデルの階層的性質とそのデータ構造を説明している。階層的モデルは、階層の各層で別個の変換マトリクスを提供できるため、変換マトリクスを変更することによってモデルまたはモデルの一部を移動することが可能である。これによって無変更モデル幾何形状（オブジェクト座標中）をアニメーションの移動オブジェクトとして用いることができる。

連想メモリ

大半の連想メモリ (CAM) は、入力ベクトルと、CAMに保存する各データ

ワードとの間でビット毎の一致テストを行う。この種のCAMは、すべてのワードの対応するビットが一致テストに影響しないよう、しばしばビット位置のマスキングを行う。このタスクを行うには多数のクロックサイクルが必要であるため、一致テスト中のCAMで大きさの比較を行うのは非効率的である。

CAMは現在、一部のコンピュータのバーチャルメモリーシステム内で翻訳ルックアサイド・バッファに用いられている。CAMはまた高速コンピュータネットワークでアドレスの一致にも用いられている。CAMは実用的な先行技術レンダラーでは用いられていない。

マグニチュード比較CAM (MCCAM) は、本書において、保存したデータを数として扱い、データの算術マグニチュード比較（すなわち、より小さい、より大きい、より小さいまたは等しい等）を並列して実行するCAMと定義される。これは、保存したデータを数としてではなく厳密にビットベクトルとして扱う通常のCAMとは対照的である。参照によって本書に含まれるMCCAM特許は、Jerome F. Duluk Jr. による1991年2月26日付与の米国特許番号第4, 996, 666号、「完全並列マグニチュード比較の可能な連想メモリスシステム」（以下、Duluk 特許と称する）である。特に参照するDuluk 特許内の構造には、プレフィックス「Duluk 特許」を含む（例えば、「Duluk 特許MCCAMビット回路」等）。MCCAMは先行技術のレンダラーでは用いられていない。

MCCAMの基本的内部構造はワードに編成されたメモリビットのセットで、ここでは各ワードが保存されたデータと入力データとの間で1以上の算術マグニチュード比較を実行することができる。

一般に、MCCAMでは、数のベクトルをワード配列に平行して適用した時、全ワード中のすべての算術比較が平行して発生する。このような並列検索比較オペレーションを保存データの「照会」と呼ぶ。

本発明は、ソーティング実行能力を含む各種特徴を付加することによってMC CAMのレイパビリティを強化する。この新しいMCCAMを、ソーティング・マグニチュード比較CAM (SMCCAM) と呼ぶ。

発明の概要

コンピュータグラフィックスは、コンピュータを使って図または画像を生成する芸術および科学である。このような図の生成はしばしばレンダリングと呼ばれる。例えば3次元アニメーションでのモーションの発現は、画像シーケンスを表示することによって達成する。インタラクティブ3次元(3D)コンピュータ・グラフィックスでは、ユーザはリアルタイムで自分のビューポイントを変えたり幾何形状を変えたりすることができるため、レンダリングシステムはリアルタイムにオンザフライ式に新しい画像を作成する必要がある。そのため、カラーの高品質画像でのリアルタイムのパフォーマンスがますます重要になっている。

本発明の装置および方法は、検索に加えてデータのソーティングの可能なデータシフティング能力を提供する。また、矩形形状検索領域に基づく空間データベースのシンプル検索または範囲検索を実行する新しい方法を提供する。範囲検索は、特殊な新ソーティング・マグニチュード比較連想メモリ(SMCCAM)装置で行うことができる。このSMCCAMはマグニチュード比較連想メモリを提供し、ここではメモリの各ワードのフィールドに保存したデータを

、隣接するワードの対応するフィールドにシフトすることができ、このシフトがマグニチュード比較結果に基づいて条件付きで実行される。

3Dグラフィックス方法では空間データベースに多角形スパンのパラメータを保存し、照会オペレーションをデータベースについて実行してどのスパン、またはスパン部分が可視かを決定する。スパンの空間データベースをSMCCAMに保存できる。SMCCAM装置は、そのオペレーションが高速で、標準や新しいビデオ技術(NTSC、PAL、HDTV等)、ビジュアルフライトシミュレーション、バーチャルリアリティ画像生成等のリアルタイムオペレーションをサポートできるため、重要である。

範囲検索を用いるため、理想的「閉塞領域」の保守的近似を導入する。閉塞テストによって理想的閉塞領域の近似としての閉塞テスト領域を定義する。カレントスパンポーションを新しいスパンと比較する際、ルールベース方法を用いて、新しいスパンは閉塞テストによって発見し、メモリに保存したデータ構造から読み出す。そのメモリはSMCCAMであることが有利だが、従来の空間データベ

ース方法で空間データ構造をインプリメントする従来のメモリでもよい。比較結果に基づき、考えられるいくつかの単純なルールの1つを選択し、これを適用してカレントスパンのどの片が可視かを決定する。いくつかの代替えルールを説明する。

セグメントスパン、台形スパンおよび四辺形スパンを含む複数の代替えスパン表現を設ける。これらスパンのそれぞれを独自のスパンパラメータセットで定義するが、その中にそれぞれ複数の選択肢がある。スパンパラメータはメモリのデータ構造に保存され、SMCCAMによってインプリメントすることができ、照会検索オペレーション中に用いる。これらスパンタイプにはそれぞれ利点がある

。例えば、台形スパンまたは四辺形スパンは、セグメントスパンより優れたアンチエイリアシング性能を持つ。この性能上の利点は、特定のラスタライン内の多角形の左および右辺情報を保存することによって達成する。台形および四辺形スパンの照会オペレーションを取り扱う方法について説明する。

空間検索およびソーティング装置および方法は、空間検索およびソーティングを用いてフロント・ツー・バックまたはバック・ツー・フロント順で透明な多角形スパンを正しくレンダリングできる。

この方法および装置に可能な改良として、2つの別個のメモリ（バケットソートメモリおよびカレント多角形メモリ）を使って多角形の幾何形状情報がスクリーン座標に翻訳された時にこれを保存するのではなく、1つのメモリ（ソートメモリ）を使って多角形の幾何形状情報を保存することができる。これは、カレント多角形メモリをソートメモリへのポイントのリストに代えることによって、バケットソートメモリからカレント多角形メモリに多角形パラメータをコピーする必要をなくすことにより行う。

台形スパンと四辺形スパンに考えられるもう1つの改良として、空間データ構造がスパン全体のパラメータを保存することが可能で、スパンをデータ構造から読み取った時、サブスパンのセットをオンザフライで生成し、これらサブスパンをフルスパンの高さに縦に配分し、各サブスパンがセグメントスパンとなる。カレントスパンポーションと新しいスパンを比較するためのルールはサブスパン毎

に適用し、ラスタライン内の追加多角形辺情報を与えて、これがアンチエイリアシングとなる。ここでも、リアルタイムパフォーマンスのため、空間データ構造をSMCCAMに保存することができる。

本発明の装置、システムおよび方法はまた、基本的システムにい

くつかのオプションと強化方法を与える。例を挙げると、これにはレンダリング順の保存、シングルバッファのフレームバッファを利用して従来のビデオカードとの互換性を高める、特定の多角形辺のアンチエイリアシングを選択的にオンまたはオフする能力、多角形ではなく辺ペアを保存することによって下流の処理を単純化する能力、カレント多角形メモリをポインタリストに置き換えることで除去する能力、クリッピングプレーン（フロント、リアおよび／または任意）を追加する能力、幾何形状をレンダリングの入力として保存するための幾何形状キャッシュの装備、スパンメモリおよびページメモリ I/Oバスアーキテクチャのページ数に関する柔軟なオプション、多角形以外の幾何学的基本立体（CSG基本立体等）のサポート、パフォーマンスで妥協した従来のハードウェアで本発明をサポートする代替え案などがある。

図面の簡単な説明

図 1 は、3 次元空間のオブジェクトとそのビューイングプレーンへの投影、およびディスプレイスクリーンでの画像である。

図 2 は、Deering 参考文献のジェネリック・レンダリング・パイプラインである（先行技術）。

図 3 は、標準 Z バッファ方法のメソッドフロー図である（先行技術）。

図 4 は、スパン・ソーティング・レンダリング・パイプラインである。このスパンメモリはSMCCAM800でインプリメントできる。

図 5 は、スパン・ソーティング・レンダリング・アーキテクチャのブロック図である。

図 6 は、9 個の多角形がディスプレイスクリーンと交差する様子

と特定のラスタラインに含まれる様子を示す例である。

図7は、多角形スパンをモデル化する次の3つの方法である：1) ラインセグメント、2) 台形、および3) 四辺形。

図8は、ソーティング・マグニチュード比較連想メモリ800 (SMCCAM 800) 内のデータ編成である。

図9は、SMCCAMワード900ブロック図である。

図10は、2つの潜在的閉塞スパンを含むスパン閉塞テスト照会がカバーする領域である。

図11は、 $x-z$ 空間で示す1本のラスタライン上のスパンセット例である。

図12は、各スパン周囲の境界ボックスを除く図11に示すのと同じスパンセット例である。

図13は、スパン・ソーティング・レンダリング方法1300のフロー図である。

図14は、スパン・ソーティング・レンダリング方法1300の一部である多角形処理方法1400のフロー図である。

図15は、スパン・ソーティング・レンダリング方法1300の一部であるカレント多角形メモリ処理方法1500のフロー図である。

図16は、スパン・ソーティング・レンダリング方法1300の一部であるバケット・ソート・メモリ処理方法1600のフロー図である。

図17は、ライト・スパン・パラメータ方法1700のフロー図である。

図18は、スパン・ソーティング・レンダリング方法1300の一部である簡易スパン・ラスタリゼーション方法1800のフロー図である。

図19は、スパン・インタラクション命名定義である。

図20は、スパンインタラクションタイプの36のバラエティである。

図21は、スパン・ソーティング・レンダリング方法1300の一部であるセグメント・スパン・ラスタリゼーション方法2100のフロー図である。

図22は、セグメント・スパン・ラスタリゼーション方法2100の一部であるルール1 2200方法のフロー図である。

図23は、セグメント・スパン・ラスタリゼーション方法2100の一部であ

るルール2 2300方法のフロー図である。

図24は、セグメント・スパン・ラスタリゼーション方法2100の一部である
ルール3 2400方法のフロー図である。

図25は、セグメント・スパン・ラスタリゼーション方法2100の一部である
ルール4 2500方法のフロー図である。

図26は、セグメント・スパン・ラスタリゼーション方法2100の一部である
ルール5 2600方法のフロー図である。

図27は、15のスパンを持つラスタラインの例である。各スパンの境界ボックスと、x軸に沿った各スパンのxL位置を共に示す。

図28は、フェイズロックラスタライン処理とディスプレイのタイミング図である。

図29は、シングルバッファのフレームバッファのタイミング図である。

図30は、スパンラスタリゼーションが1以上のスクリーンディスプレイタイムをカバーする場合のシングルバッファのフレームバッファのタイミング図である。

図31は、スパンプロセッサのハードウェアアーキテクチャのブロック図である。

図32は、スパンプロセッサ内の比較およびオフセット計算ブロックのブロック図である。

図33は、先行技術のMCCAM CMOSビットセルである。

図34は、SMCCAMビットセルの「TTLタイプ」回路である。

図35は、マスタスレーブ・フリップフロップのマスタハーフに静的記憶装置を用いるSMCCAMビットセルのCMOS回路である。

図36は、マスタスレーブ・フリップフロップの両ハーフに動的記憶装置を用いるSMCCAMビットセルのCMOS回路である。

図37は、SMCCAMビットセルの配列である。

図38は、ラスタライン内の縦の複数スパンである。

図39は、図示の台形スパンが共に動作しなかった場合、可視の遠いスパンを

閉塞するため共に動作する 2 個の台形スパンである。

図 4 0 は、 $x-z$ 空間で示す台形スパンセットの例である。

図 4 1 は、上部および下部別処理のための S O T 照会である。

図 4 2 は、スパン上部とスパン下部に別個に閉塞処理を行う台形スパン・ラスタライゼーション方法バージョンを示す台形スパンセットの例である。

図 4 3 は、各可視性トランジションを処理するための S O T 照会である。

図 4 4 は、複雑な形状を持つ S O T 照会である。

図 4 5 は、広い検索領域を持つ S O T 照会である。

図 4 6 は、可視不透明面を太い黒線で、可視半透明スパンを太い陰付き線で示したラスタラインのセグメントスパンセットである。

図 4 7 は、エンドポイントあたり 1 つの z 値のみ使う近似台形ス

パンの四辺形スパンへの変化である。

図 4 8 は、 $x-z$ 空間で示す四辺形スパンセットの例である。

図 4 9 は、ダイレクトスパン生成によるスパン・ソーティング・レンダリング・パイプラインである。

図 5 0 は、ダイレクトスパン生成によるスパン・ソーティング・レンダラー・アーキテクチャである。

図 5 1 は、ソートメモリ 4 9 0 4 に保存される多角形パラメータの代替えセットである。

図 5 2 は、ジェネリック三角形パラメータである。

図 5 3 は、スパンパラメータレジスタ 3 1 0 4 に保存される代替えスパン表現である。

図 5 4 は、ラスタライズプロセッサ 5 1 2 に送られる代替えスパン表現である。

図 5 5 は、スパンメモリ 4 0 8 に保存されるスパンパラメータの代替えセットである。

図 5 6 は、角を含める場合生成できる複数の台形スパンである。

発明の詳細な説明

スパン・ソーティング・レンダリング・パイプライン

図 4 は、スパン・ソーティング 3 D グラフィックス・パイプライン 4 0 0 を示すが、最初の 6 つのステップを多角形処理 1 4 0 0 ステップとして定義する。最初の 5 つのステップ（変換 2 0 2、クリップテスト 2 0 4、面決定 2 0 6、ライティング 2 0 8、および透視分割 2 1 2）は、ジェネリック 3 D グラフィックス・パイプライン 2 0 0 の最初の 6 つのステップの内の 5 つと同じである。スパン・ソーティング 3 D グラフィックス・パイプライン 4 0 0 はオブジェクト精密で動作し、座標はディスプレイスクリーン 1 0 4 の領域

に限定されないため、クリップ 2 1 0 ステップは省略される。クリップ 2 1 0 ステップの省略によって、ビューボリューム 1 0 6 中のすべての多角形を同様に扱うことができる。しかしながら、クリップ 2 1 0 ステップは所望により挿入することができる。スパン・ソーティング 3 D グラフィックス・パイプライン 4 0 0 は、専用ハードウェアで構築するか、完全にソフトウェアで行うか、またはこの 2 つを組み合わせることができる。以下、専用ハードウェアのインプリメンテーションを想定し、新しい装置を説明する。

新しい装置のトップレベルのブロック図を図 5 に示す。多角形処理 1 4 0 0 ステップは、多角形プロセッサ 5 0 2 で実行する。多角形プロセッサ 5 0 2 の最後のステップは、多角形パラメータ 4 0 2 の計算である。多角形プロセッサ 5 0 2 が出力する多角形パラメータは、スパン・ソーティング 3 D グラフィックス・パイプライン 4 0 0 の残りが必要とする多角形を記述する。

各多角形について、多角形パラメータがバケット・ソート・メモリ 4 0 0 に書き込まれ、次のものを含む： 1) それぞれ、 (x_1, y_1, z_1) 、 (x_2, y_2, z_2) 、 (x_3, y_3, z_3) 等と定義される多角形頂点 V_1 、 V_2 、 V_3 等のオブジェクト精密スクリーン座標での位置、 2) Gouraud（または Phong）シェーディングおよび／またはテクスチャマッピングパラメータのための頂点カラーなどを含むカラー情報、 3) 各辺の y に対する x および z の辺の導関数（すなわち $\delta x / \delta y$ および $\delta z / \delta y$ ）、 4) 開始ラスタライン（または最上）、 y_{START} 、 5) 終了ラスタライン（または最下）、 y_{END} 、および 6) 変数 d に割り当てられ

た x に対する z のスパン導関数（すなわち、 $\delta x / \delta y$ ）。このリストの最後の4つを計算し402、これは多角形プロセッサ502で行う。

多角形を三角形と想定し、辺の導関数を次のように計算する：

$$\begin{array}{lll} \frac{\delta x_{12}}{\delta y_{12}} = \frac{x_1 - x_2}{y_1 - y_2} & \frac{\delta x_{23}}{\delta y_{23}} = \frac{x_2 - x_3}{y_2 - y_3} & \frac{\delta x_{31}}{\delta y_{31}} = \frac{x_3 - x_1}{y_3 - y_1} \\ \frac{\delta z_{12}}{\delta y_{12}} = \frac{z_1 - z_2}{y_1 - y_2} & \frac{\delta z_{23}}{\delta y_{23}} = \frac{z_2 - z_3}{y_2 - y_3} & \frac{\delta z_{31}}{\delta y_{31}} = \frac{z_3 - z_1}{y_3 - y_1} \end{array} \quad (\text{式 1})$$

開始ラスタライン、 y_{START} と終了ラスタライン、 y_{END} を計算する時、一般にディスプレイスクリーン104の辺を考慮しなければならない。図6に、8個の多角形（この場合、三角形）を有するディスプレイスクリーン104を示すが、この一部がディスプレイスクリーン104の辺と交差している。多角形がディスプレイスクリーン104の右または左辺と交差しない場合、 y_{START} および y_{END} は次のように計算できる：

$$\begin{array}{l} y_{\text{START}} = \text{MinOf}(y_{\text{MAX}}, \text{IntPartOf}(\text{MaxOf}(y_1, y_2, y_3, \dots))) \\ y_{\text{END}} = \text{MaxOf}(0, \text{IntPartOf}(\text{MinOf}(y_1, y_2, y_3, \dots))) \end{array} \quad (\text{式 2})$$

y_{START} の計算でまず多角形の頂点の y 座標の最大値がわかり、多角形の「最上」 y 座標が見つかる。そして関数 $\text{IntPartOf}()$ が最上 y 座標「の整数部」を取り、多角形の「最上」 y 座標のラスタラインを計算する。多角形のスタートはディスプレイスクリーン104の最上ラスタラインより上にはあり得ないため、多角形が非存在ラスタラインからスタートする場合、 $\text{MinOf}()$ 関数は y_{MAX} を代用する。 y_{END} の計算も同様に行う。図6に、 y_{MAX} からスタートする多角形602と、ラスタラインゼロで終了する別の604を示す。

多角形がディスプレイスクリーン104の右または左辺と交差すると、 y_{START} および y_{END} は、ディスプレイスクリーン104内で多角形に影響を受ける最初で最後のラスタラインでなければならない。図6において、例の多角形606は $y_A 608$ で始まり、 y

$B 610$ で終わる。別の例の多角形612は $y_C 614$ で始まりラスタラインゼロ

で終わる。ディスプレイスクリーン 1 0 4 の右および左辺の効果は無視できるが、スパン生成 1 5 0 0 でディスプレイスクリーン 1 0 4 内にないスパンが生成されるため、これをテストしてこのようなスパンを放棄しなければならない。

スパン導関数 d (すなわち $\delta z / \delta x$) の計算は、従来の様々な方法の 1 つを使ってできる。

$$d = \frac{\delta z}{\delta x} \quad (\text{式 3})$$

スパン導関数は各プレーナ多角形について一定で、正または負の値を取ることができる。

バケット・ソート・メモリ 4 0 4 への書き込みは各多角形の y_{START} に従って行う。1 本のラスタラインにつき 1 つの「バケット」があり、多角形はその開始ラスタラインに対応する 1 つのバケットに入れる。各バケット内で多角形をソートする必要はない。バケット・ソート・メモリ 4 0 4 中のデータ管理は、バケット・ソート・プロセッサ 5 0 4 が行い、これはバケット毎に 1 つのリストをリンクさせてソートすることができる。さらに、バケット・ソート・メモリ 4 0 4 はダブルバッファとして、シーンに関連する書き込みオペレーションを前のシーンからの読み取りオペレーションと同時に実行することができる。

前述のカラー情報すべてをバケット・ソート・メモリ 4 0 4 に保存したら、これをスパン・ソーティング・レンダラー 5 0 0 (図 5 参照) を介してラスタライズ・プロセッサ 5 1 2 に運ばなければならない。あるいは、バケット・ソート・メモリ 4 0 4 のカラー情報を情報ポインタで置き換えて、別の多角形情報メモリ 5 1 4 に保存することができる。情報ポインタは多角形情報メモリ 5 1 4 へのア

ドレスで、ここには多角形のすべてのカラー情報が保存され、ラスタライズ・プロセッサ 5 1 4 が用いてカラー情報を読み取ることができる。情報ポインタはラスタライズ・プロセッサ 5 1 2 に運ばれる。情報ポインタは多角形のすべてのカラー情報と比べるとほとんどビットがないため、これによってスパン・ソーティング・レンダラー 5 0 0 内の高価なハードウェアメモリを節約できる。この節約は次の理由から可能である：1) カラー情報は多角形全体を通じて一定で、多角

形のすべてのスパンが共有できる、2) カラー情報は、ラスタライズ・プロセッサ512が可視スパンポーションをカラー化するまで必要ない。本書後半では情報ポインタと多角形情報メモリ514の利用を想定する。

スパン・ソーティング・レンダラー500内で、現在レンダリングしているラスタラインに影響するすべての多角形のリストをカレント多角形メモリ406に保持する。そのため、シーンのレンダリングが始まる時、カレント多角形メモリ406は空である。レンダリング中、特定のラスタラインに到達すると、そのラスタライン（すなわちそのバケット）に y_{START} を持つすべての多角形がバケット・ソート・メモリ404から読み出され、カレント多角形メモリ406に付加される。レンダリングが進むにつれ、多角形の y_{END} 値に到達するので多角形はカレント多角形メモリ406から削除される。カレント多角形メモリ406のデータは、カレント多角形プロセッサ506が管理し、これはパイプライン図400に示すように、「バケット・ソート・メモリ404のデータ処理」オペレーション1600を実行する。また、カレント多角形プロセッサ506は、ラスタラインに影響する多角形をスパン生成プロセッサ508に送る。

ラスタラインのレンダリング開始から、そのラスタラインにスバ

ンを持つ各多角形がスパン生成プロセッサ508に入力され、これがスパン生成1600を実行する。スパン生成プロセッサ508は、式1の辺導関数を含む各多角形の幾何学的特性を用いて、カレントラスタライン内のそのスパンの左右エンドポイントの、オブジェクト精密スクリーン座標での位置を計算する。スパンは、図7に示すように、単純なラインセグメント、台形または四辺形にモデル化できる。スパンの幾何学的特性は照会プロセッサ510に送られ、これら特性には次のものが含まれる：1) スパンの左エンドポイントの座標、2) スパンの右エンドポイントの座標、3) スパン導関数 d 、および4) 情報ポインタ。スパンのエンドポイントの座標は次のいずれかになる：1) スパンを図7Aに示すラインセグメントとしてモデル化した時に用いられるカレントラスタライン内の (x, y) ポイント、2) スパンを図7Bに示す台形としてモデル化した時に用いられるカレントラスタラインの上辺と下辺の2つの (x, z) ポイント、または3

）スパンを図7Cに示す四辺形としてモデル化した時に用いられるカレントラスタラインの上辺と下辺の z 座標と x 座標を含む (x_u, x_l, z) トリプレット。図6に示す8個の多角形例の内、616、618、620、622の4個はカレントラスタライン例640上にスパン626、628、630、632を有する。スパン例の1つ626は負の x 値で始まっていることに注意する。

あるラスタラインについて、スパンの幾何学的特性がスパン生成プロセッサ508から出力されると、これらを照会プロセッサ510が受け取ってすべてをダブルバッファのスパンメモリ406のバンクの1つに書き込む。スパンがスパンメモリ406に書き込まれると、これらは次によって優先順にソートされる：1）左 x 座標、2）左 z 座標、および3）スパン導関数。一方のバンクへのスパン

データの書き込みと同時に、スパンメモリ406の他方のバンクのスパンデータを処理して可視スパンポーションを発見する。

照会プロセッサ510がスパンメモリ406の一方のバンクに保存されたスパンの処理を完了し、次のラスタラインのすべてのスパンをスパン生成プロセッサ508から受け取って他方のバンクに書き込むと、スパンメモリ406の2つのバンクがスワップされる。バンクスワップ後、照会プロセッサ510とスパンメモリ406は共同してスパンデータに関する算術比較（以下、照会オペレーションと称する）と各種算術演算を実行して1800または2100を処理し、どのスパン、またはどのスパンポーションがシーンで可視かを決定する。これら可視スパン（またはスパンポーション）はラスタライズ・プロセッサ512に送られる。照会プロセッサ510とスパンメモリ406のより詳細なブロック図を図31に示す。

ラスタライズ・プロセッサ512は完全に可視なスパン（またはスパンポーション）のみ受け取る。各スパンを処理するため、ラスタライズ・プロセッサ512はインクレメンタルなスパンレンダー412をセットアップしてから、スパン補間220を実行する。ピクセルカラー値は、多角形情報メモリ514、および場合によってテクスチャマップメモリ516に保存されたデータを利用して生成する。ピクセルカラー値を生成すると、これらはラスタラインメモリ416に書

き込まれる。

ラスタラインメモリ416は、複数の完全なラスタラインについてすべてのカラーピクセル値を保存できる。すなわち、416は現在（ラスタライズ・プロセッサ512を介して）デジタルアナログコンバータ228に送られているラスタラインと複数の後続ラスタラインについてカラー値を保存する。あるラスタラインのすべての値がデジタルアナログコンバータ228に送られると、ラスタライ

ンメモリ416の対応する部分に他のラスタラインを上書きすることができる。このように、必要なメモリの総量は、先行技術の3Dグラフィックスレンダーにはダブルバッファされたフルディスプレイスクリーン104のフレームバッファ224が必要なのに比べて、ラスタライン数行（最小で2行）で済む。これによってメモリの多くのメガバイトを節約できる。ラスタラインメモリ416は、ラスタリゼーションプロセスがデジタルアナログコンバータ228「より先に動作」するため複数のラスタラインのデータを保存し、処理に平均時間以上を必要とするラスタラインにある程度の時間クッションを与えることができる。

数行のラスタライン分のメモリしか持っていないことで、スパン・ソーティング・レンダラー500のアーキテクチャは平均ラスタライン表示速度に遅れないようにする必要がある。そのため、非常に大きな幾何形状データベース（スパン・ソーティング・レンダラー500がついていける最大より大きい）をレンダリングしなければならない場合、ラスタラインメモリ416をダブルバッファされたフルディスプレイスクリーン104のフレームバッファ224に置き換えることができる。

ソーティング・マグニチュード比較連想メモリ800

スパンメモリ406のページ内のデータ記憶装置800の図を図8に示す。このデータ配列は、ソートされたデータ構造として典型的なランダムアクセスメモリ（RAM）に常駐することもでき、ソーティング・マグニチュード比較連想メモリ（SMCCAM）に常駐することもできる。本書後半ではSMCCAMインプリメンテーションを想定する。

SMCCAMは新しいタイプのMCCAMであり、メモリレジスタ（またはワ

ード) セットから構成され、各ワードは複数のフィー

ルドから構成され、各フィールドは、1) 整数または浮動小数点の数として数を保存することができ、2) 保存した数とすべてのワードにブロードキャストされた別の数との算術比較を実行し、3) その内容を次のワードの対応するフィールドにシフトする。スパン・ソーティング・レンダラー500内のスパンメモリ406として用いる場合、SMCCAMはスパンセットを保存し、並列検索およびソーティングオペレーションを実行して可視スパンポーションを見つける。

図8に示すように、SMCCAM800内のデータ記憶装置はSMCCAMワード900に分けられ、各ワード900はカレントラスタラインの1スパンに対応するデータを(照会オペレーション実行によって)保存・処理する。この図は、0から $W-1$ まで番号付けられた W 個のSMCCAMワード900をすべて示す。各SMCCAMワード900には、次の7個の数値フィールド(および本書で示す変数名は n 番目のワード)を含む: 1) ワード番号フィールド802、 w で、アドレスに類似の、各ワード900の固定された(すなわちリードオンリー)一意の識別番号、2) 左Xフィールド804、 X_{nL} で、スパンの左エンドポイントの x 座標を保存する、3) 左Zフィールド806、 z_{nL} で、スパンの左エンドポイントの z 座標を保存する、4) 右Xフィールド808、 X_{nR} で、スパンの右エンドポイントの x 座標を保存する、5) 右Zフィールド810、 z_{nR} で、スパンの右エンドポイントの z 座標を保存する、6) スパン導関数フィールド812、 d_n で、スパンの $\delta z / \delta x$ スロープを保存する、および7) 情報ポインタフィールド814、 i_n で、スパンの多角形のカラー情報について多角形情報メモリ514にポインタを保存する。

各SMCCAMワード900はまた次のものを保存する: 1) 有

効フラッグ816、 F_{nv} で、SMCCAMワード900が有効データを保存しているか否かを示す1つのビット値、および2) 照会フラッグ818、 F_{nq} で、SMCCAMワード900が照会オペレーションに肯定応答したか否かを示す1つのビット値。いずれのフラッグビット816、818も対応する「ワイヤードノ

ア」バスを持ち、すべてのワード 9 0 0 がそのフラッグビットをオフにしているか否かを示す。特に、有効フラッグ 8 1 6、 F_{nv} については、 F_{nv} がすべての n （すなわちすべてのワード）について偽なら、信号 AllWordInvalid 9 9 0 が表明される。同様に、 F_{nq} がすべての n について偽なら、信号 NullQueryResponse 9 9 2 が表明される。2つのワイヤードノア信号、AllWordInvalid 9 9 0 と NullQueryResponse 9 9 2 は、照会結果を外部コントローラ（照会プロセッサ 5 1 0 中にある）にフィードバックし、方法内での進め方に関する決定（すなわち「ブランチ」）を行うメカニズムとなる。

フィールドおよびフラッグ 8 0 4 から 8 1 8 までの命名には、フィールドサブスクリプトの第 1 部（例： z_{3L} がワード 3 中）としてのワード番号フィールド 8 0 2 の値を含む。本書後半で、スパン・ソーティング・レンダラー 5 0 0 に各種機能を追加するため各ワードのフィールド数が増える。図 8 に、2つの可変ワードインデックス、 n および w を示し、これは本書を通じて SMCCAM ワード 9 0 0 の参照に用いる。

SMCCAM ワード 9 0 0 のブロック図を図 9 に示すが、ここに 7 個のフィールド 8 0 2 から 8 1 4 のそれぞれを示す。7 個のフィールド、 x_{wL} 8 0 4、 z_{wL} 8 0 6、 x_{wR} 8 0 8、 z_{wR} 8 1 0、 d_w 8 1 2 および i_w 8 1 4 はそれぞれ、配列バスセット 9 1 0：バス W 9 1 2、バス X L 9 1 4、バス Z L 9 1 6、バス X R 9 1 8、バス Z R 9 2 0、バス D 9 2 2 及びバス 1 9 2 4 内に対応するデータ

バスを有する。フィールド 8 0 2 から 8 1 2 の 6 個はメモリ 9 0 2 または 9 0 4 におよびコンパレータ 9 0 6 または 9 0 8 に分かれる。ワード番号フィールド 8 0 2 に 1) 固定メモリ 9 0 2、および 2) 一致比較回路 9 0 6 が含まれ、これは照会オペレーションで用いられて固定メモリ値 9 0 2 を入力バス、バス W 9 1 2 のデータと比較する。フィールド 8 0 4 から 8 1 2 の 5 個は、1) 読み取り／書き込みメモリ 9 0 4 および 2) 算術比較回路 9 0 6（より小さい等のテスト）を含み、これは照会オペレーションで用いられて保存されたメモリ値 9 0 4 を対応する入力バスのデータと比較する。7 番目のフィールド、情報ポインタフィールド 8 1 4、 i_w は、一般に照会オペレーションに参加しない単なる読み取り／書

き込みメモリである。

照会オペレーションは、SMCCAMワード900の配列800の検索、ソート、読み取り、書き込みに用いられる。照会オペレーションは、照会データを配列バス910を介してすべてのワード900に供給することで、すべてのSMCCAMワード900が平行して実行する。SMCCAM800は照会論理配列850を含み、これは照会論理930回路のセットで、SMCCAMワード900にそれぞれ1個ある。

平行して、各SMCCAMワード900内では、照会オペレーション結果が照会論理930によって計算される。照会論理930は、コンパレータ906および908の結果と、有効フラッグ816および照会フラッグ818の値を受け取り、これらの値に選択的ブール演算を行って (QueryCntrl932で選択) 照会結果ビットを生成し、この照会結果ビットを有効フラッグ816または照会フラッグ818のいずれかに書き戻す。

照会オペレーションを実行すると、各ワード900は照会結果を

生成し、これはその900有効フラッグ816か照会フラッグ818に保存される。すべての有効フラッグ816のセットは有効フラッグワード830と呼ばれ、すべての照会フラッグ818のセットは有効フラッグワード834と呼ばれる。検索を行うと、照会結果はどのワードが照会オペレーションのパラメータを満たすか指定する。照会オペレーションの例は次の通りである：

$$F_{nQ} = F_{nV} \wedge (x_{nL} \leq x_{CL}) \quad (\text{式 4})$$

但し、次が発生する：1) x_{CL} が、バスZL916を介してすべてのワード900にブロードキャストされる、2) 各ワード900の左Xフィールド804が、その804の内容、 x_{nL} をバスZL916の値と比較することで、 $(x_{nL} \leq x_{CL})$ を実行する。3) 各ワード900の照会論理930が、そのワード900の照会結果である式4の右辺を実行する、4) 各ワード900において、照会結果が式4の左辺に示す照会フラッグ818に保存される。および5) 照会フラッグ818がすべてのワード900で偽なら、信号、NullQueryResponse 992が表

明される。

ブール演算の追加例を式5に示すが、ここで x_{CL} 、 S_P 、 x_{CR} 、 z_{CF} 、 z_{CL} および z_{CR} は、配列バス910を介してすべてのSMCCAMワード900にブロードキャストされたデータで、 n はオペレーションが発生する特定のワード900である（これはすべての n 、すなわちすべてのSMCCAMワード900について発生する）。

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \\
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{CL}) \vee (z_{nR} < z_{CL})] \quad (\text{式 5}) \\
 F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{CR}) \wedge (x_{nL} = z_{CR}) \\
 F_{nV} &= F_{nV} \wedge (x_{nR} > x_{CR}) \\
 F_{nQ} &= \overline{F_{nV}} \vee (x_{nL} > x_{nL}) \vee [(x_{nL} = x_{nL}) \wedge (x_{nL} > z_{nL})] \vee [(x_{nL} = x_{nL}) \wedge (x_{nL} = z_{nL}) \wedge (d_n > d_n)]
 \end{aligned}$$

照会オペレーションメカニズムはSMCCAM800に保存されたデータについて固有に検索オペレーションを実行し、検索結果は照会フラッグ818に保存される。データをSMCCAM800に書き込む必要がある場合、次のシーケンスが発生する：1) 有効フラッグワード830（または一部アプリケーションでは照会フラッグワード834も使える）を選んで、書き込むワード900を決定する、2) 有効フラッグワード830の内容を830に対応する優先順位レジスタ840に入力する、3) 優先順位レジスタ840は、フラッグワード830内で論理「ゼロ」の最初の出現を発見し、これが最初の無効ワードとなる、4) 論理「ゼロ」の最初の出現のあるSMCCAMワード900が選択されたワード900で、配列バス910のデータをそのワード900に保存する、そして5) 選択したワード900はその有効フラッグ816を表明し、そのワードに有効なデータが入っていることを示す。

読み取りオペレーションも、次のシーケンスで同様に動作する：1) 有効照会フラッグワード834を選んで、読み取るワード900を決定する、2) 照会フラッグワード834の内容を834に対応する優先順位レジスタ840に入力する、3) 優先順位レジスタ840は、フラッグワード834内で論理「1」の最初の出現を発見する、4) 論理「1」の最初の出現のあるSMCCAMワード900が選択されたワード900で、その900の内容を配列バス910のデータに出力する、5) 選択されたワード900はその照会フラッグ818をデアサー

ションし、そのワードは内容を読み取られたことを示し、後続読み取り（介入照会なし）が他のワードを読み取る、そして6）照会フラッグ818がすべてのワード900で偽なら、信号、NullQueryResponse 992を表明し、他のワードが読み取りオペレーションに応答しないことを示す。照会、書き込み

、読み取りオペレーションは、Duluk 特許の説明にやや類似して動作する。

SMCCAMワード900内の優先順位レゾルバ840の部分を実行順位レゾルバセクション934と呼び、これら934は次のバスを介してSMCCAMワード900全体で通信する：ValidPriOut[w-1]936、QueryPriOut[w-1]937、ValidPriOut[w]938およびQueryPriOut[w]939。または、優先順位レゾルバ840はツリー状構造で構築し、最悪回路伝搬をWではなくlog Wに比例して遅らせることもできる。

SMCCAM800はまた、Duluk 特許で説明する装置にはないソートオペレーションも実行する。前述のように、スパン・ソーティング・レンダラー500は、スパンをスパンメモリ406に書き込む時に次に従ってスパンをソートする：1）左x座標、2）左z座標、および3）スパン導関数。SMCCAM800では、スパンを挿入しなければならないスパンリスト中の場所を位置づけることでソーティングを行い、この場所からSMCCAM800の内容を1つのワード900にシフトダウンし、新しいスパンを空になったワード900に書き込む。

SMCCAM800は、データをワード毎にシフトしてソートしたスパンリストに場所を設けることができる。ソートオペレーションは次のシーケンスで行われる：1）書き込むスパンの（前述）幾何学的特性を照会オペレーションのためにSMCCAM800に入力する、2）照会フラッグ818に書き込んだ照会結果で、各SMCCAMワード900で式6の照会オペレーションを実行し、無効データが入っているか、スパン書き込み中のスパンリストで後でソートしなければならないスパンの入っているすべてのワードを識別する、3）照会フラッグワード834の内容を対応する優先順位レ

ゾルバ840に入力する、4）優先順位レゾルバ840が照会フラッグワード8

34内で論理「1」の最初の出現を発見し、選択ワード900を決定する、5) 選択ワード900と選択ワード900の後のすべてのSMCCAMワード900について、保存したデータを次のワード900にシフトする、そして6) 選択したワード900で配列バス910のデータをワード900に保存する。

(式6)

$$F_{AQ} = \overline{F_{AV}} \vee (x_{AL} > x_{AL}) \vee [(x_{AL} = x_{AL}) \wedge (z_{AL} > z_{AL})] \vee [(x_{AL} = x_{AL}) \wedge (z_{AL} = z_{AL}) \wedge (d_s > d_s)]$$

フィールド802から814のデータのSMCCAMワード900の1つから次へのシフトは、前のワード900のシフトアウトバス970に接続されたシフトインバス950を介して行う。シフトインバス950のリストには次のものが含まれる: ShiftOutXL[w-1]952、ShiftOutXR[w-1]954、ShiftOutZL[w-1]956、ShiftOutZR[w-1]958、ShiftOutD[w-1]960、およびShiftOutI[w-1]962。シフトアウトバス970のリストには次のものが含まれる: ShiftOutXL[w]972、ShiftOutXR[w]974、ShiftOutZL[w]976、ShiftOutZR[w]978、ShiftOutD[w-1]980、およびShiftOutI[w]982。本書で用いる回路信号命名法には、「[w]」などの括弧入りインデックスの利用が含まれ、これがバス配列の一部であることを示す(「C」コンピュータ言語の配列命名法に類似)。ここで「[w]」はw番目のワード900、「[w-1]」はワード800セットの直前のワード900を示す。

スパン閉塞テスト

図10に、ディスプレイスクリーン104の1ラスタラインに対応する、x-zプレーンのラインセグメント(すなわち「セグメントスパン」)で表す3つのスパン、1002、1004、1006を示す。スパンの1つ1002は、対応する閉塞領域1008と共

に示す。領域を閉塞1008するスパンがないと、対応するスパン1002は隠れない。閉塞領域1008の形状は台形であるため、他のスパンがその1008内にあるか否かを決定する計算は高価になり、領域1008に対して多くのスパンをテストしなければならない場合は特に高価である。閉塞領域1008の単純化近似が、矩形スパン閉塞テスト領域1010(以下SOT領域1010と称す

る)である。SOT領域1010は、閉塞テスト中のスパン(またはスパンポーション)から生成し、以下これをカレントポーション、Sc1004と称する。エンドポイント(x_{CL} 、 z_{CL})1012および(x_{CR} 、 z_{CR})1014のカレントポーションSc1004について、エンドポイント(x_{nL} 、 z_{nL})および(x_{nR} 、 z_{nR})のスパン1006がSOT領域1010内にあるか否かの決定は次のように行う:

SOT領域中= (式7)

$$(x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < \text{MaxOf}(z_{CR}, z_{CL})) \vee (z_{nR} < \text{MaxOf}(z_{CR}, z_{CL}))]$$

式7に必要な算術計算は4つの比較と1つの「最大選択」関数(式で必要なこの2つは同一)である。SMCCAM800が式7の演算を実行するには、SMCCAM800外で「最大選択」を行って次を決定する:

$$z_{CF} = \text{MaxOf}(z_{CR}, z_{CL}) \quad (\text{式8})$$

そして値 x_{CL} 、 x_{CR} および z_{CF} を配列バス910に入力して、該当するフィールド804から810で比較を行う。このように、SMCCAM800に保存されるすべてのスパンは平行してテストされ、ナノ秒の内に閉塞スパンを検索する。本書において、SOT照会が見つかるスパンはすべて閉塞スパンと呼ばれる。

スパン閉塞テスト照会(以下、SOT照会と称する)の完全な式

を式9として示す。

$$F_{nQ} = F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \quad (\text{式9})$$

完全なSOT照会式は、論理積に次のあと2つの項を追加する: 1) 無効データのあるワードが自身を誤って閉塞スパンを保存すると識別しないために必要な有効フラッグ816、および2) S_P がカレントポーションに対応するワード番号フィールド802の場合、スパンが自身を閉塞と識別するのを防ぐために行うテスト、 $n \neq S_P$ 。完全なSOT照会式は、照会結果を各SMCCAMワード900の照会フラッグ818に保存する。SOT照会結果は、信号、NullQueryResponse 992で利用でき、表明された値は閉塞スパンが見つからなかったことを

示す。

閉塞領域1008の近似としてSOT領域を用いることで、保守的なエラーが導入される。すなわち、SOT照会で、SOT領域1010内にあるが、実際はカレントポーションの背後にある閉塞スパンを見つける可能性がある。但し、SOT照会はカレントポーションを実際に閉塞するスパンを見つけ損なうことはない。

スパン・ソーティング・レンダリング方法1300

SOT照会が、カレントポーションを実際には閉塞しない閉塞スパンを不注意で発見するには、カレントポーションの境界ボックス（次のセクションで説明）が閉塞スパンの境界ボックスと重なっていないなければならない。そのため、スパン・ソーティング・レンダリング方法1300の最もシンプルな形式では、スパンの境界ボックスが重ならないと想定して、実際にはカレントポーションを閉塞しない閉塞スパンを扱わなくてよいようにしている。この想定はほとんどのレンダリングアプリケーションについて実際的ではないが、方法1300の説明の一部を単純化する。そのため、この簡易（実

際的ではないが）バージョンを先に説明する。後のセクションで、方法1300を境界ボックスの重なるスパンの処理に拡大する。

図11に、同じラスタラインの14個のスパンセット（ S_0 1100から S_{13} 1113まで番号付け）を示すが、ここで各スパンはラスタラインの $x-z$ 平面（ y 座標は固定）のラインセグメントで表される。スパンの可視部分を太いラインで、隠れた部分を細いラインで示す。例えば、 S_4 1104と S_6 1106は隠れており、 S_0 1110は2つの可視ポーションを持つ。スパンは左エンドポイントの x 座標位置に従ってソートされている。そのため、スパンはSMCCAM800に保存された時割り当てられたワード番号フィールド802に従って左から右に番号付けられている。例えば、スパン S_4 1104はSMCCAMワード900番号4に保存され、左エンドポイントはポイント（ x_{4L} 、 z_{4L} ）に、右エンドポイントは（ x_{4R} 、 z_{4R} ）にある。

図12に、同じ14個のスパンを示すが、各スパンは境界ボックスで囲まれて

いる。図11と図12のスパンのセットは、スパン境界ボックスが重ならないように選んでいる。図12のスパンを使って、スパン・ソーティング・レンダリング方法1300の最もシンプルな形式を説明するが、このバージョンは、スパン境界ボックスが重ならないと想定する簡易スパン・ラスタリゼーション方法1800を含む。

スパン・ソーティング・レンダリング方法1300を、本書記載の装置のコンテキストで説明するが、これは汎用コンピュータのソフトウェアでも実行できる。図13はスパン・ソーティング・レンダリング方法1300のトップレベル方法図である。この方法1300は、ジェネリック3Dグラフィックス・パイプライン200が図3の方法フロー図300を実行するのと同じ方法で、図4のスパ

ン・ソーティング・レンダリング・パイプライン400によって実行される。

スパン・ソーティング・レンダリング方法1300では、シーンが生成され1302（ワールド座標のオブジェクト位置等）、各シーン内で多角形が取得され1304、スパン・ソーティング・レンダリング・パイプライン400に入力されて処理される1400。

図14に、多角形処理ステップ1400を示す。ここには、前に説明した変換202、クリップテスト204、面決定206、ライティング208、透視分割212、および多角形パラメータ計算402ステップが含まれる。これらステップの後、多角形は各自の y_{START} パラメータに従ってバケット・ソート・メモリ404に書き込まれる。

図13に戻り、あるシーンのすべての1306多角形が処理され1400、バケット・ソート・メモリ404に書き込まれると、各ラスタラインが処理される。変数、Rを使って現在のラスタライン番号を記録し、ゼロに初期化1308する。最初のラスタラインを処理するには、カレント多角形メモリ406をクリアし1310、カレントラスタラインに多角形がないことを示さなければならない。また、スパンメモリ408をクリアし1312、カレントラスタラインにスパンがないことを示さなければならない。これらオペレーション1310、1312は、メモリ406、408の内容全体を無効とマークすることで実行できる。

最初のラスタラインでは、カレント多角形メモリ406に多角形がないため、次のステップ1600でカレントラスタラインで始まる多角形をバケット・ソート・メモリ404からカレント多角形メモリ406に移し、これら多角形のスパンを作る。第2のラスタライン（すなわち、 $R=1$ ）から、カレント多角形メモリ406には恐らく多角形があるので、こ

れら多角形についてスパンを作る1500必要がある。

図15は、スパン・ソーティング・レンダリング方法1300内のカレント多角形メモリ406処理ステップ1500である。カレント多角形メモリ406の各1502多角形を読み取り1504、1506その多角形がカレントラスタラインに含まれていなければ、カレント多角形メモリ406から削除1508して、後続ラスタラインがこれを考慮しなくて済むようにする。1506その多角形が現在のラスタラインに含まれる場合、その多角形についてカレントラスタライン内のスパンとそのスパンパラメータ（前述）を生成1510する。そしてスパンパラメータをスパンメモリ408に書き込む1700。

図16は、スパン・ソーティング・レンダリング方法1300内のバケット・ソート・メモリ404処理ステップ1600である。もし1602カレントラスタラインについてバケット・ソート・メモリ404からさらに読み込む多角形があれば、そのような各多角形について、1) バケット・ソート・メモリ404から多角形を読み取る1604、2) 多角形をカレント多角形メモリ406に書き込む1606、3) 多角形はカレントラスタラインにスパンを持ち、スパンパラメータ（前述）をそのスパンについて生成する1510、および4) スパンパラメータをスパンメモリ408に書き込む1700。

スパンメモリ408にスパンパラメータを書き込む時1700、ソートオペレーションを行う。ソートオペレーションは、SMCCAM800が照会オペレーション（式6）および前述の特別書き込みオペレーションとして行う。SMCCAM800の装置はこのステップ1700を平行して行うが、図17のフロー図1700には、順次検索プロセスのように記述されている。SMCCAMワード

900セットへのインデックスとして用いるカウンタ、 n をゼロに初期化する1702。フロー図は、新しいスパンを n 番目のワード900に挿入しなければならないか否かをテストする、6つの条件に分けた1704式6の照会オペレーションを示す。かかる挿入ポイントが見つかるまで n を増分することによって1706、ワード900を進める。挿入ポイントが見つかり、カウンタ n は増分を中止し、 $W-1$ から n （カウンタ m を用いる）のすべてのワードの内容を次のワードへ転送し1708、スパンパラメータをワード n に書き込む1710。ラスラインの W スパンの合計より多くあり、ワード900の総数を超える場合、最後のワード900（すなわち、ワード $W-1$ ）に達するテスト1712を含めて例外1714条件を検出する。固定数のワード900内で作業する戦略は本書後半で提示する。

再び図13に戻り、ラスライン R のディスプレイスクリーン内のすべてのスパンをスパンメモリ408に書き込んだら、可視スパンポーションを識別する1800または2100。方法のこの部分のバージョンに簡易スパン・ラスタリゼーション方法1800があるが、これはスパン境界ボックスが重ならないと想定している。この想定を行わない他のバージョンは、以下のセクションで説明するセグメント・スパン・ラスタリゼーション方法2100と台形スパン・ラスタリゼーション方法である。

簡易スパン・ラスタリゼーション方法1800

図18に示す簡易スパン・ラスタリゼーション方法1800は、次の3つの変数セットを維持する：1) カレントラスラインの内今までどれだけレンダリングしたかを示すのに用いるカレント左 x 座標位置、 x_{CL} 、2) x_{CL} の最前スパンを記録するのに用いる現在のスパン S_p で、そのスパンパラメータ x_{PL} 、 z_{PL} 、 x_{PR} 、 z_{PR} 、

d_p 、 i_p およびワード番号を含む、および3) スパンメモリ408から最も最近読み取ったスパンを記録するのに用いる新しいスパン S_N で（一般に照会で見つける）、そのスパンパラメータ、 x_{NL} 、 z_{NL} 、 x_{NR} 、 z_{NR} 、 d_N 、 i_N およびワード番号を含む。簡易スパン・ラスタリゼーション方法1800では、カレントポ

ーション、 S_c は常に現在のスパン S_p に等しい。

ラスタライゼーション方法1800を、図12のスパンセットを例に用いて説明する。本セクション後半において、方法1800をスパン例処理と共にステップバイステップで追う。

方法1800は、1802 x_{cl} をゼロにセットし（これによってディスプレイスクリーンの左辺で始まる）、変数 S_p と S_N を無効と宣言1804することで始まる。 S_p が無効1804で、スパンメモリ408にまだ保存された有効1806スパンがあるため、 x_{CL} （すなわちゼロ）の最前スパンか、 x_{cl} にスパンがない場合はスパンメモリ408の最左有効スパンのいずれかを見つけるために検索1808を行わなければならない。

この検索1808の第1ステップは、照会1810を行って、 x_{cl} （すなわちゼロ）を含むスパンメモリ408のすべての有効スパンを見つけることである。図12を見ると、 x 座標ゼロにはスパンがない1812ことがわかる（照会1810で何も見つけない）。 x 座標ゼロにスパンがない1812ため、スパンメモリ408の最左スパン（すなわち、スパンはソートされているため最初の有効スパン）を読み取り1818、 S_p を S_0 と等しくする1200。 x_{cl} （すなわちゼロ）には有効スパンがなかったため、最左スパンは x_{ol} （ S_0 1200の左エンドポイント）まで開始せず、ゼロから x_{ol} までの範囲は全くスパンがないため、背景を x_{cl} （すなわちゼロ）から x_{ol} までレンダリング1820する。レンダリングを行っ

たので x_{cl} の値が更新され1822、 x_{pl} （すなわち x_{ol} ）になり、このポイントまでレンダリングが行われたことを示す。現在のスパン（すなわち S_0 1200）についてSOT照会を行い1824、閉塞スパンがあればこれを見つけ、他のすべてのスパン1201から1213が S_0 1200のSOT領域にあるためこれらを見つける。これら見つかったスパンの最初のものを読み取り1824、新しいスパン、 S_N の値を S_1 1201に等しくする。

方法1800はディスプレイスクリーンの右辺（すなわち、 $x_{cl} < X_{max}$ ）に達しておらず1826、 S_p と S_N の両方とも有効1804および1828で、次

のステップ1830では、 x_{CL} (すなわち、 x_{OL}) から x_{NL} (すなわち、 x_{IL}) まで S_P (すなわち、 $S_0 1200$) をレンダリングする。SOT照会1824が $S_0 1200$ の前に最左閉塞スパンを見つけており、この閉塞スパンの左の $S_0 1200$ のあらゆる部分が可視でなければならないため、このレンダリングが可能となる。このステップ1830も x_{CL} の値を x_{NL} (すなわち、 x_{IL}) に更新し、 S_N (すなわち、 $S_1 1201$) を S_P の新しい値とする。SOT照会は、閉塞スパンを見つけるため現在のスパン (すなわち、 $S_1 1201$) について行い1832、スパン $S_2 1202$ のみを見つけるが、これ1202が $S_1 1201$ のSOT領域の唯一のスパンであるためである。このスパンを読み取り1832、新しいスパン S_N の値を $S_2 1202$ に等しくする。

スパン $S_1 1201$ を $S_0 1200$ と同じ方法で処理する。方法1800はディスプレイスクリーンの右辺 (すなわち、 $x_{CL} < X_{MAX}$) に達しておらず1826、 S_P と S_N の両方とも有効1804および1828で、 S_P (すなわち、 $S_1 1201$) を x_{CL} (すなわち、 x_{IL}) から x_{NL} (すなわち、 x_{2L}) までレンダリング183

0する。 x_{CL} の値を x_{NL} (すなわち、 x_{2L}) に更新し1830、 S_N (すなわち、 $S_2 1202$) を S_P の新しい値にする。現在のスパン (すなわち、 $S_2 1202$) にSOT照会を行うが1832、 $S_2 1202$ を閉塞するものがないので、この照会ではスパンが見つからず、新しいスパン S_N の値は無効にセットされる。

ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、方法1800の次の部分1834は S_P の隣接スパン検索を含む。隣接スパンとは、左エンドポイントがカレントポーション (または簡易スパン・ラスタリゼーション方法1800では、現在のスパン) の右エンドポイントと同じポイントにあるスパンである。図12において、隣接するスパンペアの3つの例には次のものがある：1) $S_2 1202$ と $S_5 1205$ 、2) $S_5 1205$ と $S_7 1207$ 、および3) $S_6 1206$ と $S_9 1209$ 。3Dオブジェクトは一般に辺を共有する多角形のセットとして記述されるため、隣接ス

パンはしばしば発生する。そのため、高度にモザイク化されたオブジェクトでは、大半のパンが少なくとも1つの隣接するパンペアの一部となる。

方法1800のポーション1834は、 x_{CL} (すなわち、 x_{2L}) から x_{PR} (すなわち、 x_{2R}) までの S_P (すなわち、 S_2 1202) のレンダリング1836から始まる。SOT照会で閉塞パンが見つかっていないため、 S_P (すなわち、 S_2 1202) が全く隠れておらず、レンダリングしなければならないためにこれが可能である。また、 x_{CL} を x_{PR} (すなわち、 x_{2R}) にセットする1836。次に、照会を行って1838、 S_P (すなわち、 S_2 1202) の隣接パンを見つけるが、これでパン S_5 1205が見つかる。1840で隣接パンが見つかったため、隣接パン S_5 1205を読み取り1846、 S_P に割り当てる。現在のパン (すなわ

ち、 S_5 1205) にSOT照会を行うが1848、 S_5 1205を閉塞するものがないので、この照会ではパンが見つからず、新しいパン S_N の値を無効にセットする。

現在のパン、 S_5 1205を S_2 1202と同様に処理する。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、 x_{CL} (すなわち、 x_{2R}) から x_{PR} (すなわち、 x_{5R}) まで S_P (すなわち、 S_5 1205) をレンダリング1836し、 x_{CL} を x_{PR} (すなわち、 x_{5R}) にセット1836する。隣接するパンについて照会を行い1838、 S_7 1207を見つける。隣接するパン S_7 1207を読み取り1846、 S_P に割り当てる。現在のパン (すなわち、 S_7 1207) にSOT照会を行い1848、 S_8 1208のみ見つけ、これを読み取り1832、 S_N の値を S_8 1208に等しくする。

現在のパン、 S_7 1207を S_1 1201と同様に処理する。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P と S_N は共に有効で1804および1828、 x_{CL} (すなわち、 x_{5R}) から x_{NL} (すなわち、 x_{8L}) まで S_P (すなわち、 S_7 1207) をレンダリング1830する。 x_{CL} の値を x_{NL} (すなわち、 x_{8L}) に更新し1830、 S_P を S_N (すなわち、 S_2 1202) にセット

する。現在のスパン（すなわち、 S_8 1208）にSOT照会を行うが1832、この照会ではスパンが見つからず、新しいスパン S_N の値を無効にセットする。

現在のスパン、 S_8 1208を S_2 1202および S_5 1205と同様に処理する。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、 x_{CL} （すなわち、 x_{8L} ）から x_{PR} （すなわち、 x_{8R} ）まで S_P （すなわち、 S_8 1208）をレンダリング1836し、 x_{CL} を x_{PR} （すな

わち、 x_{8R} ）にセット1836する。隣接するスパンについて照会を行い1838、 S_{10} 1210を見つける。隣接するスパン S_{10} 1210を読み取り1846、 S_P に割り当てる。現在のスパン（すなわち、 S_{10} 1210）にSOT照会を行うが1848、この照会ではスパンが見つからず、新しいスパン S_N の値を無効にセットする。

現在のスパン、 S_{10} 1210を処理する時、隣接スパンの検索でそのようなスパンを見つけない。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、 x_{CL} （すなわち、 x_{8R} ）から x_{PR} （すなわち、 x_{10R} ）まで S_P （すなわち、 S_{10} 1210）をレンダリング1836し、 x_{CL} を x_{PR} （すなわち、 x_{10R} ）にセット1836する。隣接するスパンについて照会を行うが1838、隣接するスパンは見つからない。隣接するスパンが見つからなかったため1840、検索を行って x_{CL} （すなわち、 x_{10R} ）の最前スパンを見つけないといけない。しかしながら、この検索を行う前に、照会を行って完全に x_{CL} （すなわち、 x_{10R} ）の左にあるすべてのスパンを無効にする1842。 x_{CL} （すなわち、 x_{10R} ）までのカレントラスタラインをレンダリングしており、 x_{CL} （すなわち、 x_{10R} ）の左のスパンがレンダリングされているか隠れている（あるいは両方）ため、この無効化1842が可能である。無効化1842で、 S_1 1201から S_{10} 1210の有効フラッグ816を切り、 S_0 1200と S_{11} 1211から S_{13} 1213までのみ有効保存スパンとして残す。現在のスパン S_N の値と、新しいスパン S_N の値はいずれも無効に

セットする1844。

ディスプレイスクリーンの右辺に達しておらず1826、 S_P が無効で1804、まだ有効なスパンがあり1806、方法の次の部

分1808には、 x_{CL} (すなわち、 x_{10R}) の最前スパンの検索を含む。まず、照会1810を行って、 x_{CL} (すなわち、 x_{10R}) を含むすべてのスパンを見つけ、2つのスパン、 S_0 1200と S_{11} 1211が見つかる。スパンが見つかったため1812、これら2つのスパン1200と1211を読み取り1814、スパンの x_{CL} (すなわち、 x_{10R}) の z 座標を計算し、比較して、 S_{11} 1211が最前スパンであると決定する1814。 z 座標計算は、スパン導関数(スパン導関数フィールド812から)をスパン左エンドポイントからの外挿に用いることで簡略化する。最前スパン S_{11} 1211を現在のスパン、 S_P とする1814。現在のスパン(すなわち、 S_{11} 1211)にSOT照会を行うが1824、この照会でスパンが見つからず、新しいスパン S_N の値を無効にセットする。

現在のスパン、 S_{11} 1211を、 S_2 1202、 S_5 1205および S_8 1208と同様に処理する。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、 x_{CL} (すなわち、 x_{10R}) から x_{PR} (すなわち、 x_{11R}) まで S_P (すなわち、 S_{11} 1211) をレンダリング1836し、 x_{CL} を x_{PR} (すなわち、 x_{11R}) にセット1836する。隣接するスパンについて照会を行い1838、 S_{12} 1212を見つける。隣接するスパン S_{12} 1212を読み取り1846、 S_P に割り当てる。現在のスパン(すなわち、 S_{12} 1212)にSOT照会を行うが1848、この照会ではスパンが見つからず、新しいスパン S_N の値を無効にセットする。

現在のスパン、 S_{12} 1212を、 S_{10} 1210と同様に処理する。ディスプレイスクリーンの右辺に到達しておらず1826、 S_P は有効で1804、 S_N は無効で1828、 x_{CL} (すなわち、

x_{11R}) から x_{PR} (すなわち、 x_{12R}) まで S_P (すなわち、 S_{12} 1212) をレンダリング1836し、 x_{CL} を x_{PR} (すなわち、 x_{12R}) にセット1836する

。隣接するスパンについて照会を行うが 1838、隣接するスパンは見つからない。隣接するスパンが見つからなかったため 1840、照会を行って完全に x_{cl} (すなわち、 x_{12R}) の左にあるすべてのスパンを無効にし 1842、 S_{11} 1211 と S_{12} 1212 の有効フラッグ 816 を切り、 S_0 1200 と S_{13} 1213 のみ有効保存スパンとして残す。現在のスパン S_N の値と、新しいスパン S_N の値はいずれも無効にセットする 1844。

ディスプレイスクリーンの右辺に達しておらず 1826、 S_P が無効で 1804、まだ有効なスパンがあり 1806、方法の次の部分 1808 には、 x_{cl} (すなわち、 x_{12R}) の最前スパンの検索を含む。まず、照会 1810 を行って、 x_{cl} (すなわち、 x_{12R}) を含むすべてのスパンを見つけ、1つのスパン、 S_0 1200 のみ見つかる。スパンが見つかったため 1812、これを現在のスパン、 S_P にする 1814。現在のスパン (すなわち、 S_0 1200) に SOT 照会を行い 1848、 S_{13} 1213 のみを見つけ、これを読み取り 1832、 S_N の値を S_{13} 1213 に等しくする。

ディスプレイスクリーンの右辺に到達しておらず 1826、 S_P と S_N は共に有効で 1804 および 1828、 x_{cl} (すなわち、 x_{12R}) から x_{NL} (すなわち、 x_{13L}) まで S_P (すなわち、 S_0 1200) をレンダリング 1830 する。 x_{cl} の値を x_{NL} (すなわち、 x_{13L}) に更新し 1830、 S_P を S_N (すなわち、 S_{13} 1213) にセットする。現在のスパン (すなわち、 S_{13} 1213) に SOT 照会を行うが 1832、この照会ではスパンが見つからず、新しいスパン S_N の値を無効にセットする。

ディスプレイスクリーンの右辺に達しておらず 1826、 S_P が有効で 1804、 S_N が無効で 1828、 x_{cl} (すなわち、 x_{13LL}) から x_{PR} (すなわち、 x_{13R}) まで S_P (すなわち、 S_{13} 1213) をレンダリング 1836 し、 x_{cl} を x_{PR} (すなわち、 x_{13R}) にセット 1836 する。隣接するスパンについて照会を行うが 1838、隣接するスパンは見つからない。隣接するスパンが見つからなかったため 1840、照会を行って完全に x_{cl} (すなわち、 x_{13R}) の左にあるすべてのスパンを無効にし 1842、 S_0 1200 と S_{13} 1213 の有効フ

ラッグ816を切り、最後に有効なスパンを取り除く。現在のスパン S_N の値と、新しいスパン S_N の値はいずれも無効にセットする1844。

ディスプレイスクリーンの右辺に達しておらず1826、 S_P が無効で1804、有効なスパンがない1806。そのため、 x_{CL} （すなわち、 x_{13R} ）から x_{MAX} （すなわち、ラスタラインの右辺）まで背景をレンダリングする。これでカレントラスタライン体がレンダリングされたので、図13の方法フロー図に示すように次のラスタラインを処理できる。

方法1800はスパンの可視ポーションをレンダリングしており、完全に隠れているスパン（すなわち、 S_3 1203、 S_4 1204、 S_6 1206および S_9 1209）を完全に無視していることに注意する。深さ複雑性の大きいシーンでは、スパンの大きな部分が同様に無視されるため、大きな計算上の節約となる。本方法1800の独自の特徴は、隣接スパンが共に動作して後ろのスパンを閉塞する能力である。

方法1800の別の重要な特徴は、ラスタライン内のピクセルカラー化処理が、ラスタラインの全スパンのピクセル数でなく、ラスタラインのピクセル数にほぼ比例していることである。これは、ス

パン内の隠れたポーションは決してピクセルカラー化プロセスに送られないためである。例えば、右のスパンがピクセルの境界内で終わることがあるため、ピクセルが2つのスパンに影響されるので、ピクセルカラー化処理は、ラスタライン内のピクセル数に「ほぼ」比例する。この場合、2つのスパンからのカラーがブレンドされてそのピクセルの最終カラーを形成する。

スパン・インタラクション・タイプとスパン・インタラクション・パラメータ

前述のように、SOT照会はSMCCAM800がハードウェア中で実行できるだけ単純な比較オペレーションセットである。しかしながら、スパン境界ボックスが重なる一般的なケースでは、SOT照会でカレントポーションを閉塞しないスパンを発見することがある。さらに、SOT照会はカレントポーションと交差するスパンを見つけることがあり、いずれも部分的に可視となる。セグメント・スパン・ラスタリゼーション方法2100は、カレントポーションとSOT領

域中のスパンのインタラクションを、多くのスパン・インタラクション・タイプ（以下、SITと略す）の1つに分類し、SITに基づくレンダリング規則を適用することによってこの欠点を解決する。SITは、2つのスパンのスパン・インタラクション・パラメータ（以下、SIP）から決定する。

セグメント・スパン・ラスタリゼーション方法2100が進む間、2100は次の4つのスパンとその関連パラメータを保持する：1) 前述の x_{CL} の最前スパンである現在のスパン、 S_P 、2) 前述の S_P のSOT領域中のスパンである新しいスパン、 S_N 、3) 現在のスパン S_P のサブセクションであるカレント・ポーシヨン、 S_C 、および4) S_N を一時的にセーブして、 S_P のSOT領域でより多くのスパンを読みとれるようにするため1つの深さスタックと

して用いるセーブされたスパン、 S_S 。

スパン、スパンのエンドポイント、SIPおよびSITの命名については、図19に例として示す。現在のスパン、 S_P 1902は、エンドポイント（ x_{PL} 、 z_{PL} ）1904、1906と（ x_{PR} 、 z_{PR} ）1908、1910を持ち、破線楕円に囲まれ、黒実線と細点線の和集合として示される。 S_P 1902の黒実線サブセクションはカレントポーシヨン S_C 1912で、エンドポイント（ x_{CL} 、 z_{CL} ）1914、1916と（ x_{CR} 、 z_{CR} ）1918、1920を持つ。 S_C 1912に対応するSOT領域1922は破線に囲まれる。新しいスパン S_N 1932は、エンドポイント（ x_{NL} 、 z_{NL} ）1934、1936と（ x_{NR} 、 z_{NR} ）1938、1940を持ち、境界ボックス1942に囲まれる。

スパン・インタラクション・パラメータ（すなわち、SIP）は、4つの x 座標値、特に2つのスパンのエンドポイント： x_{NL} 1934、 x_{NR} 1938、 x_{CL} 1914および x_{CR} 1918における、 S_C 1912と S_N 1932の z 座標の算術比較結果である。詳しく言うと、4つのスパン・インタラクション・パラメータ（すなわち、SIP）とは次の通りである：1) S_C 1912は x_{NL} 1934にポイントを持たないため、例で「非存在」と示される x_{NL} 1934から z_{NL} 1936の S_C 1912の z 座標の比較、2) S_C 1912上のポイントは z_{NR} 1940より遠い（すなわち、 z 座標が大きい）ため、例で「より遠い」と示される

$x_{NR}1938$ から $z_{NR}1940$ の S_c1912 の z 座標の比較、3) S_N1932 上のポイントは $z_{CL}1916$ より遠い(すなわち、 z 座標が大きい)ため、例で「より遠い」と示される $x_{CL}1914$ から $z_{CL}1916$ の S_N1932 の z 座標の比較、および4) S_N1932 は $x_{CR}1918$ にポイントを持たないため、例で「非存在」と示される $x_{CR}19$

18から $z_{CR}1920$ の S_N1932 の z 座標の比較。4つのSIPはそれぞれ次の値の1つを取ることができる: 1) 「非存在」(「非」と略す)、2) 「より遠い」(「遠」と略す)、3) より近い「近」と略す)、または4) 「等しい」。図19の例では、4つのSIPは、図に示すように非、遠、遠、非である。この特定の4セットはSIT23で、本書後半で説明するようにルール4を呼び出す。

SIPは、除法などの高価な計算を避け、計算しやすいように選んだ。比較で用いる z 座標の計算は次のように行う:

$$\begin{aligned} x_{NL} \text{ での } S_c \text{ の } z \text{ 座標} &= z_{CL} + (x_{NL} - x_{CL}) d_c \\ x_{NR} \text{ での } S_c \text{ の } z \text{ 座標} &= z_{CL} + (x_{NR} - x_{CL}) d_c \\ x_{NL} \text{ での } S_N \text{ の } z \text{ 座標} &= z_{NL} + (x_{CL} - x_{NL}) d_N \\ x_{CR} \text{ での } S_N \text{ の } z \text{ 座標} &= z_{NL} + (x_{CR} - x_{NL}) d_N \end{aligned} \quad (\text{式10})$$

図20は、考えられる49のSITすべてを一覧したものである。例えば、図20CはSIT3を示し、 S_N とペアの S_c の3例を示すが、ここでSIPは近、等、非、非である。SIPに可能な組み合わせはそれぞれ1つのSITに対応する。セグメント・スパン・ラスタリゼーション方法2100中にあるSITに遭遇すると、複数のルールの内1つが呼び出されるが、図20の各SITはそれが呼び出すルールを明示する。SIT19やSIT22など一部SITは、方法2100で遭遇することがあり得ないため、何のルールも呼び出されず、タグ「不可」が図20に示される。

セグメント・スパン・ラスタリゼーション方法2100

図21に示すセグメント・スパン・ラスタリゼーション方法2100は、ラインセグメント(図7Aに示す)で表すスパンを処理する。この方法は、図13に

示すスパン・ソーティング・レンダリング方法1300内で利用することができる。本書のこのセクション

では、図21から図26の方法フロー図を極めて一般的な用語で説明する。次のセクションでは、スパン例の複雑なセットを使って、方法2100の機能をステップバイステップで説明する。

セグメント・スパン・ラスタライゼーション方法2100は、2102と2104変数の初期化で始まる。方法2100は、ルール1 2200からルール5 2600と呼ばれるサブ方法呼び出す。各ルールは次のことを考慮する：1) もしあれば、 S_c のどの部分をレンダリングすべきか、2) S_p および/または S_c の更新、3) S_s の更新、4) スパンメモリ408に保存されたスパンの無効化、5) SOT照会実行、および/または6) S_N の更新。

ルール1 2200 (図22に示す) は、 S_p が無効の場合に呼び出す。2200は x_{cl} の最前スパンを探し、最前スパン S_p を作り、SOT照会を実行し、 S_N を更新する。

ルール2 2300 (図23に示す) は、 S_c に潜在的閉塞スパンが残っていない場合に呼び出す。そのため、2300は現在のスパン、 S_p を x_{cl} から x_{cr} までレンダリングする。そして S_c を次の先行順でセットする：1) セーブしたスパン、 S_s 、2) 隣接スパン、または3) 完全に x_{cr} の左のすべてのスパン無効化後、最初に有効なスパン。いかなる場合も、 x_{cl} は x_{cr} にセットされ、 S_s は無効とされる。

ルール3 2400 (図24に示す) は、カレントポーション、 S_c が x_{nl} で閉塞されるが、 x_{cl} から x_{nl} まで可視であるようなSITの場合に呼び出す。そのため、現在のスパン、 S_p を x_{cl} から x_{nl} までレンダリングする。そして、新しいスパン、 S_N を現在のスパンにして、その処理を始める。 S_c は x_{cr} までレンダリングしていないので、セーブした S_s を無効にする。

ルール4 2500 (図25に示す) は、 S_c が S_N と交差し、

交点が可視であるようなSITの場合に呼び出す。カレントポーション、 S_c と

新しいスパン、 S_N との間の (x_i, z_i) における交点を計算する。 S_c は x_i の右に閉塞されているので、 x_{cr} を x_i に変更して S_c を修正する。新しいスパン、 S_N を、ルール1 2200で後で必要となる場合に備えてセーブしたスパン、 S_s として保存する。そして次の潜在的閉塞スパンを読み取り、 S_N とする。

ルール5 2600 (図26に示す)は、カレントポーシオン、 S_c が新しいスパン、 S_N を閉塞して、 S_N のどれも x_{cl} から x_{cr} まで可視でない場合に呼び出す。そのため、 S_N を放棄し、次の潜在的閉塞スパンを読み取り、 S_N とする。このルールを適用している時は S_c をレンダリングしないので、セーブしたスパン、 S_s の状態は無変化のままである。

セグメント・スパン・ラスタライゼーション方法2100の詳細な例

セグメント・スパン・ラスタライゼーション方法2100をステップバイステップで完全に説明するため、図27の15のスパン例セットを使って、方法2100の動作方法を示す。この例のスパンは、セグメント・スパン・ラスタライゼーション方法2100のすべての部分を実行できるように選んだ。典型的なシーンの典型的なラスタラインに比べると、この例は、境界ボックスの重なりとスパンの交差の部分が大きいため、異常に複雑である。セグメント・スパン・ラスタライゼーション方法2100の各ステップを説明する際、方法フロー図の対応する参照指示番号をステップ説明の冒頭に一覧する。ステップ説明において、照会オペレーションの式はまず方法フロー図に現れる通りに示してから、2行目で、同じ式をすべての変数に図27の例の値を代入して示す。ステップ説明ではまた、代入文をカスケードして(複数の「=」記号によって)、完全代入変数

への到達方法を示す。

図27の例では、セグメント・スパン・ラスタライゼーション方法2100が始まる時、SMCCAM800に S_0 から S_{14} と指定された15の有効スパンがある。これらスパンをSMCCAMワード900の0から14に保存するため、それぞれが($F_{nv}=1$)を持ち、SMCCAMワード900が有効な内容を持つことを示す。SMCCAMワード900の残り、すなわちワード15からW-1は、それぞれ($F_{nv}=0$)を持ち、SMCCAMワード900が無効な内容を持つ

ことを示す。故に、セグメント・スパン・ラスタリゼーション方法2100のスタートにおいて、有効フラグワード840、Fvは有効スパンセットを指定し、このセットは空ではない。

図27に示す15のスパン例セットについて、セグメント・スパン・ラスタリゼーション方法2100を使ったステップバイステップの処理は次のようになる：

1) 2102 カレントポーションの左x値、 x_{cl} は、レンダリングが現在のラスタラインをどのくらい遠くまで進んだかを示す。故に、ラスタラインのスタートで設定($x_{cl}=0$)となる。カレントポーションの左z値、 z_{cl} を無限にセットし、背景の前にあらゆるスパンを見込む。背景の深さ位置は無限と考える。背景は多角形から構成されず、アプリケーションを考え、カラーブラックにセットする。カレントポーションの右のxおよびy値、 x_{cr} と y_{cr} は、有効な現在のスパン、 S_p がないため、「無効」にセットする。これは代入文によって次のように記述される：

$$\begin{aligned} x_{cl} &= 0 ; & x_{cr} &= \text{無効} ; \\ z_{cl} &= \text{無限} ; & z_{cr} &= \text{無効} ; \\ z_{cf} &= \text{無限} ; \end{aligned} \quad (\text{式11})$$

2) 2104 SMCCAM800外部でスパン情報を保存するレジスタ内のデータ（すなわち、 S_p 、 S_n 、および S_s ）も無効と指定する：

$$\begin{aligned} S_p &= \text{無効} ; \\ S_n &= \text{無効} ; \\ S_s &= \text{無効} ; \end{aligned} \quad (\text{式12})$$

3) 2106、2108、および2200 現在のスパン、 S_p が有効でなく、Fvが空でないため（少なくとも何らかのスパンが有効であることを示す）、ルール1を呼び出す。ここで、ルール1を用いて第1のスパンを探し、現在のスパン、 S_p を作る。

3A) 2202 照会を行って、 x_{cl} で（すなわち、x座標ゼロ）カレントラスタラインに影響を与えそうな有効スパンすべてを探す。照会は、各SMCCA

Mワード900で次のように実行する：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (x_{nL} \leq x_{CL}) \\ F_{nQ} &= F_{nV} \wedge (x_{nL} \leq 0) \end{aligned} \quad (\text{式13})$$

この照会でスパンが見つからないため、セット F_Q は空である。

3 B) 2204、2206および2208 F_Q が空で S_P が有効でないため、 x_{CL} でカレントラスタラインに影響する有効スパンがあってはならない。そのため、 F_V の第1スパンを読み取り、現在のスパン、 S_P とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_P &= \text{読み取り}(n) = 0 ; \\ x_{PL} &= \text{読み取り}(x_L) = x_{0L} ; \quad x_{PR} = \text{読み取り}(x_R) = x_{0R} ; \\ z_{PL} &= \text{読み取り}(z_L) = z_{0L} ; \quad z_{PR} = \text{読み取り}(z_R) = z_{0R} ; \quad (\text{式14}) \\ d_P &= \text{読み取り}(d) = d_0 ; \quad i_P = \text{読み取り}(i) = i_0 ; \end{aligned}$$

スパンはSMCCAM800で左から右の順で保存されているため、 F_V からの第1スパンの読み取りで、最左有効スパン、 S_0 が

与えられる。

3 C) 2210 x_{CL} (すなわち、 x 座標ゼロ) でカレントラスタラインに影響を与えそうな有効スパンがないので、最左有効スパンは x_{PL} (すなわち、 x_{0L}) までスタートせず、背景を x_{CL} から x_{PL} (すなわち、0から x_{0L}) までレンダリングしなければならない。

可能な代替え方法として、次の場合は背景のレンダリングを避けることができる：1) 多角形が常にビューイング面の100%をカバーするとわかっている、または2) 各ラスタラインの第1スパンの処理前に、ラスタバッファを背景カラーで初期化する。

3 D) 2212 これで現在のスパン、 S_P (すなわち、 S_0) があるので、次のようにセットすることで、カレントポーション、 S_C を S_P に等しくセットする：

$$\begin{aligned}
 x_{CL} &= x_{PL} = x_{OL}; & x_{CR} &= x_{PR} = x_{OR}; \\
 z_{CL} &= z_{PL} = z_{OL}; & z_{CR} &= z_{PR} = z_{OR}; \\
 CF &= \max(x_{PL}, x_{PR}) = \max(z_{OL}, z_{OR}) = z_{OL};
 \end{aligned}
 \tag{式15}$$

カレントポーシヨン、 S_c は S_o に等しい。カレントポーシヨンの遠Z値、 z_{CF} を、「最大を探す」ファンクションで z_{OL} にセットする。

3 E) 2 2 1 4 これまで定義された現在のスパン、 S_p (すなわち、 S_o) と定義されたカレントポーシヨン、 S_c (すなわち、 S_o) があるので、カレントポーシヨン、 S_c (すなわち、 S_o) を閉塞する可能性のあるスパンについて検索を行う。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \\
 F_{nQ} &= F_{nV} \wedge (n \neq 0) \wedge (x_{nL} < x_{OR}) \wedge (x_{nR} > x_{OL}) \wedge [(z_{nL} < z_{OL}) \vee (z_{nR} < z_{OL})]
 \end{aligned}
 \tag{式16}$$

この照会で、カレントポーシヨン、 S_c (すなわち、 S_o) につ

いてすべての潜在的閉塞スパンのセットを作る。照会で S_1 から S_{14} が見つかる。一般に、検索オペレーションに用いる照会には、論理積で F_{nV} を含み、SMCCAMワード900に保存された無効データが F_o にビットを生じ、誤って表明されないようにする。

3 F) 2 2 1 6 および 2 2 1 8 F_o が空でないため、カレントポーシヨン、 S_c (すなわち、 S_o) を閉塞する可能性のあるスパンがなければならない。そのため、 F_o の第1スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 1;$$

$$\begin{aligned}
 x_{NL} &= \text{読み取り}(x_L) = x_{1L}; & x_{NR} &= \text{読み取り}(x_R) = x_{1R}; \\
 z_{NL} &= \text{読み取り}(z_L) = z_{1L}; & z_{NR} &= \text{読み取り}(z_R) = z_{1R}; \\
 d_N &= \text{読み取り}(d) = d_1; & i_N &= \text{読み取り}(i) = i_1;
 \end{aligned}
 \tag{式17}$$

F_o の第1スパンの読み取りで、セットの第1スパン、 S_1 が与えられる。読み取りプロセスではまた、($F_{1o} = 0$) をセットすることで F_o から S_1 を削除する。

4) 2 1 0 9、2 1 0 6、2 1 1 0 および 2 1 1 2 ディスプレイスクリーン

の右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、現在のスパン、 S_P (すなわち、 S_0) も新しいスパン、 S_N (すなわち、 S_1) も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：(遠、非、非、遠)。

5) 2 1 1 4 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ 1 4 であることがわかる。

6) 2 1 1 6 および 2 4 0 0 スパン・インタラクション・タイプがタイプ 1 4 なので、ルール 3 を呼び出して次のように S_0 を処

理する：

6 A) 2 4 0 2 カレントポーシヨン、 S_c (すなわち、 S_0) が x_{NL} (すなわち、 x_{1L}) から隠れているが、 x_{NL} (すなわち、 x_{1L}) の左は可視である。そのため、現在のスパン、 S_P (すなわち、 S_0) を x_{CL} から x_{NL} (すなわち、 x_{0L} から x_{1L}) までレンダリングする。

6 B) 2 4 0 4 現在のスパンの処理が完了したので (この時点)、次のようにセットして、 S_N (すなわち、 S_1) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 1; \\ x_{PL} &= x_{NL} = x_{1L}; & x_{PR} &= x_{NR} = x_{1R}; \\ z_{PL} &= z_{NL} = z_{1L}; & z_{PR} &= z_{NR} = z_{1R}; \\ d_P &= d_N = d_1; & i_P &= i_N = i_1; \end{aligned} \quad (\text{式 18})$$

6 C) 2 4 0 6 カレントポーシヨン、 S_c を次のようにセットすることで S_N に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{NL} = x_{1L}; & x_{CR} &= x_{NR} = x_{1R}; \\ z_{CL} &= z_{NL} = z_{1L}; & z_{CR} &= z_{NR} = z_{1R}; \\ CF &= \max(z_{NL}, z_{NR}) = \max(z_{1L}, z_{1R}) = z_{1L}; \end{aligned} \quad (\text{式 19})$$

あるいは、カレントポーシヨン、 S_c は、同じ値を持っているため、現在のスパン、 S_P (S_N ではなく) にセットすることもできる。しかしながら、このステップと前のステップを平行して行えないため、この代替え方法はハードウェアインプリメンテーションには望ましくない。

6 D) 2408 セーブしたスパン、 S_s を無効化するが、この例のこの時点で既に無効だった。

6 E) 2410 新しいカレントポーション、 S_c (すなわち、 S_1) があるので、これを閉塞する可能性のあるスパンを検索する

。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 1) \wedge (x_{nL} < x_{1R}) \wedge (x_{nR} > x_{1L}) \wedge [(z_{nL} < z_{1L}) \vee (z_{nR} < z_{1L})] \end{aligned} \quad (\text{式20})$$

この照会で、カレントポーション、 S_c (すなわち、 S_1) についてすべての潜在的閉塞スパンのセットを作る。照会で S_0 と S_2 から S_{14} が見つかる。

6 F) 2412および2414 F_0 が空でないため、カレントポーション、 S_c (すなわち、 S_1) を閉塞する可能性のあるスパンがなければならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 0 ;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{0L} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{0R} ;$$

$$z_{NL} = \text{読み取り}(z_L) = z_{0L} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{0R} ; \quad (\text{式21})$$

$$d_N = \text{読み取り}(d) = d_0 ; \quad i_N = \text{読み取り}(i) = i_0 ;$$

F_0 の第1スパンの読み取りで、セットの第1スパン、 S_0 が与えられる。読み取りプロセスではまた、($F_{00} = 0$) をセットすることで F_0 から S_0 を削除する。

7) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、現在のスパン、 S_p (すなわち、 S_1) も新しいスパン、 S_N (すなわち、 S_0) も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：(遠、非、遠、非)。

8) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ23であることが

わかる。

9) 2116、2118および2500 スパン・インタラクション・タイプがタイプ23なので、ルール4を呼び出して次のようにS1を処理する：

9A) 2502 カレントポーシヨン、 S_c (すなわち、 S_i) と新しいスパン、 S_N (すなわち、 S_o) がインタラクトして、 S_c をその長さのどこか： S_c (すなわち、 S_i) と S_N (すなわち、 S_o) の間の交点で、可視から隠れに移行させる。交点を計算し、図27に示すように、 (x_1, z_1) 2702であることがわかる。

9B) 2504 カレントポーシヨン、 S_c (すなわち、 S_i) を、その右エンドポイントを交点に変更して短縮する。 S_c の右エンドポイントを次のようにセットして修正する

$$\begin{aligned} x_{CR} &= x_1; \\ z_{CR} &= z_1; \\ CF &= \max(z_{CL}, z_1) = z_{CL} = z_{IL}; \end{aligned} \quad (式22)$$

現在のスパン、 S_P はまだ S_i のすべてに等しい。

9C) 2506 新しいスパン、 S_N (すなわち、 S_o) を、セーブしたスパン、 S_s としてセーブする。これは、 S_P (すなわち、 S_i) を x_1 までレンダリングする場合、後で S_s が必要とするために行う。

$$\begin{aligned} S_s &= S_N = 0; \\ x_{SL} &= x_{NL} = x_{OL}; & x_{SR} &= x_{NR} = x_{OR}; \\ z_{SL} &= z_{NL} = z_{OL}; & z_{SR} &= z_{NR} = z_{OR}; \\ d_s &= d_N = d_0; & i_s &= i_N = i_0; \end{aligned} \quad (式23)$$

セーブしたスパン、 S_s の x_1 の右のポーシヨンが x 座標 x_1 で S_c と隣接する。方法の後半で、 S_c を x_1 までレンダリングする

場合、 S_s は現在のスパン、 S_P になる。

9D) 2508 カレントポーシヨン、 S_c (すなわち、 x_{IL} から x_1 までの S_i) を変更し、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$F_{nQ} = F_{nV} \wedge F_{nQ} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(x_{nL} < x_{CP}) \vee (x_{nR} < x_{CP})] \quad (\text{式 24})$$

$$F_{nQ} = F_{nV} \wedge F_{nQ} \wedge (n \neq 1) \wedge (x_{nL} < x_I) \wedge (x_{nR} > x_{IL}) \wedge [(x_{nL} < x_{IL}) \vee (x_{nR} < x_{IL})]$$

この照会で、カレントポーシオン、 S_c （すなわち、 x_{IL} と x_I の間の S_I ）についてすべての潜在的閉塞スパンのセットを作る。照会では S_2 しか見つからない。照会によって実行した論理積には F_0 を含み、前に読み取った（そして放棄した） S_I の潜在的閉塞スパンが F_0 に再び入れられて、不要に再処理されないようにする。

この方法のバリエーションとして、このステップを省略し、照会ではなく既存 F_0 からの読み取りを継続して F_0 からのスパンを除去するものがある。このバリエーションは、照会をなくすことで実行時間を節約するが、 F_0 のスパン数を減少できずに実行時間が増加することがある。

9 E) 2 5 1 0 および 2 5 1 2 F_0 が空でないため、カレントポーシオン、 S_c （すなわち、 x_{IL} から x_I までの S_I ）を閉塞する可能性のあるスパンがなければならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 2 ;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{2L} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{2R} ;$$

$$z_{NL} = \text{読み取り}(z_L) = z_{2L} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{2R} ; \quad (\text{式 25})$$

$$d_N = \text{読み取り}(d) = d_2 ; \quad i_N = \text{読み取り}(i) = i_2 ;$$

F_0 の第1スパンの読み取りで、セットの第1（そして唯一の）スパン、 S_2 が与えられる。読み取りプロセスではまた、 $(F_2 = 0)$ をセットすることで F_0 から S_2 を削除し、 F_0 を空にする。

1 0) 2 1 0 9、2 1 0 6、2 1 1 0 および 2 1 1 2 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、現在のスパン、 S_P （すなわち、 S_I ）も新しいスパン、 S_N （すなわち、 S_2 ）も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：（遠、非、非、近）。

1 1) 2 1 1 4 スパン・インタラクション・パラメータからスパン・インタ

ラクシオン・タイプを決定し、タイプ 1 3 であることがわかる。

1 2) 2 1 1 6 および 2 4 0 0 スパン・インタラクシオン・タイプがタイプ 1 3 なので、ルール 3 を呼び出して次のように S₁ の処理を継続する：

1 2 A) 2 4 0 2 カレントポーシオン、S_c (すなわち、x_{1L} と x₁ の間の S₁) が x_{NL} (すなわち、x_{2L}) から隠れているが、x_{NL} (すなわち、x_{2L}) の左は可視である。そのため、現在のスパン、S_p (すなわち、S₁) を x_{CL} から x_{NL} (すなわち、x_{1L} から x_{2L}) までレンダリングする。

1 2 B) 2 4 0 4 現在のスパンの処理が完了したので (この時点)、次のようにセットして、S_N (すなわち、S₂ を現在のスパン、S_p にする：

$$\begin{aligned} S_p &= S_N = 2; \\ x_{PL} &= x_{NL} = x_{2L}; & x_{PR} &= x_{NR} = x_{2R}; \\ z_{PL} &= z_{NL} = z_{2L}; & z_{PR} &= z_{NR} = z_{2R}; \\ d_p &= d_N = d_2; & i_p &= i_N = i_2; \end{aligned} \quad (\text{式 26})$$

1 2 C) 2 4 0 6 カレントポーシオン、S_c を次のようにセットすることで S_N に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{NL} = x_{2L}; & x_{CR} &= x_{NR} = x_{2R}; \\ z_{CL} &= z_{NL} = z_{2L}; & z_{CR} &= z_{NR} = z_{2R}; \\ CF &= \max(z_{NL}, z_{NR}) = \max(z_{2L}, z_{2R}) = z_{2L}; \end{aligned} \quad (\text{式 27})$$

1 2 D) 2 4 0 8 セーブしたスパン、S_s を無効化する。ステップ 9 C で、S_s の値は S₁ の値にセットされたが、ここでこれらに無効というラベルが付く。

$$S_s = \text{無効}; \quad (\text{式 28})$$

1 2 E) 2 4 1 0 新しいカレントポーシオン、S_c (すなわち、S₂) があるので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各 SMCCAM ワード 9 0 0 で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 2) \wedge (x_{nL} < x_{2R}) \wedge (x_{nR} > x_{2L}) \wedge [(z_{nL} < z_{2L}) \vee (z_{nR} < z_{2L})] \end{aligned} \quad (\text{式 29})$$

この照会で、カレントポーシオン、S_c (すなわち、S₂) についてすべての潜在的閉塞スパンのセットを作る。照会で S₀、S₃ および S₄ が見つかる。

1 2 F) 2 4 1 2 および 2 4 1 4 F₀ が空でないため、カレントポーシオン

、 S_c (すなわち、 S_2) を閉塞する可能性のあるスパンがなければならない。そのため、 F_0 の第 1 スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_N &= \text{読み取り} (n) = 0 ; \\ x_{NL} &= \text{読み取り} (x_L) = x_{0L} ; & x_{NR} &= \text{読み取り} (x_R) = x_{0R} ; \\ z_{NL} &= \text{読み取り} (z_L) = z_{0L} ; & z_{NR} &= \text{読み取り} (z_R) = z_{0R} ; & (式30) \\ d_N &= \text{読み取り} (d) = d_0 ; & i_N &= \text{読み取り} (i) = i_0 ; \end{aligned}$$

F_0 の第 1 スパンの読み取りで、セットの第 1 スパン、 S_0 が与えられる。読み取りプロセスではまた、($F_{00} = 0$) をセットすることで F_0 から S_0 を削除する。

13) 2109、2106、2110 および 2112 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、現在のスパン、 S_P (すなわち、 S_2) も新しいスパン、 S_N (すなわち、 S_0) も有効である。そのため、これら 2 つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：(非、非、遠、遠)。

14) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ 32 であることがわかる。

15) 2116、2118 および 2600 スパン・インタラクション・タイプがタイプ 32 なので、ルール 5 を呼び出して次のように S_2 を処理する：

15A) 2602 および 2604 新しいスパン、 S_N (すなわち、 S_0) が S_c (すなわち、 S_2) の x 軸投影内に隠れている。そのため、 S_N (すなわち、 S_0) を放棄する。 F_0 が空でないため、カレントポーション、 S_c (すなわち、 S_2) を閉塞する可能性のあるスパンがなければならない。そのため、 F_0 の第 1 スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_N &= \text{読み取り} (n) = 3 ; \\ x_{NL} &= \text{読み取り} (x_L) = x_{3L} ; & x_{NR} &= \text{読み取り} (x_R) = x_{3R} ; \\ z_{NL} &= \text{読み取り} (z_L) = z_{3L} ; & z_{NR} &= \text{読み取り} (z_R) = z_{3R} ; & (式31) \end{aligned}$$

$d_N = \text{読み取り}(d) = d_3; \quad i_N = \text{読み取り}(i) = i_3;$

F_0 の第1スパンの読み取りで、セットの第1スパン、 S_3 が与

えられる。読み取りプロセスではまた、($F_{30} = 0$)をセットすることで F_0 から S_3 を削除する。 S_2 はまだ現在のスパン、 S_P である。セーブしたスパン、 S_5 の状態は無変化のままなので、無効のままである。

16) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず(すなわち、 $x_{CL} < x_{max}$)で、現在のスパン、 S_P (すなわち、 S_2)も新しいスパン、 S_N (すなわち、 S_3)も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる:(遠、非、非、近)。

17) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ13であることがわかる。

18) 2116および2400 スパン・インタラクション・タイプがタイプ13なので、ルール3を呼び出して次のように S_2 の処理を継続する:

18A) 2402 カレントポーション、 S_c (すなわち、 S_2)が x_{NL} (すなわち、 x_{3L})から隠れているが、 x_{NL} (すなわち、 x_{3L})の左は可視である。そのため、現在のスパン、 S_P (すなわち、 S_2)を x_{CL} から x_{NL} (すなわち、 x_{2L} から x_{3L})までレンダリングする。

18B) 2404 現在のスパンの処理が完了したので、次のようにセットして、新しいスパン、 S_N (すなわち、 S_3)を現在のスパン、 S_P にする:

$$\begin{aligned} S_P &= S_N = 3; \\ x_{PL} &= x_{NL} = x_{3L}; & x_{PR} &= x_{NR} = x_{3R}; \\ z_{PL} &= z_{NL} = z_{3L}; & z_{PR} &= z_{NR} = z_{3R}; \\ d_P &= d_N = d_3; & i_P &= i_N = i_3; \end{aligned} \quad (\text{式32})$$

18C) 2406 カレントポーション、 S_c を次のようにセットすることで S_N (すなわち、 S_3)に等しくセットする:

$$\begin{aligned}
 x_{CL} &= x_{NL} = x_{3L}; & x_{CR} &= x_{NR} = x_{3R}; \\
 z_{CL} &= z_{NL} = z_{3L}; & z_{CR} &= z_{NR} = z_{3R}; \\
 CF &= \max(x_{NL}, x_{NR}) = \max(z_{3L}, z_{3R}) = z_{3L};
 \end{aligned}
 \tag{式33}$$

1 8 D) 2 4 0 8 セーブしたスパン、S_Sを無効化するが、これは前のステップで既に無効というラベルが付いている。

$$S_S = \text{無効}; \tag{式34}$$

1 8 E) 2 4 1 0 新しいカレントポーシヨン、S_c (すなわち、S₃)があるので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \\
 F_{nQ} &= F_{nV} \wedge (n \neq 3) \wedge (x_{nL} < x_{3R}) \wedge (x_{nR} > x_{3L}) \wedge [(z_{nL} < z_{3L}) \vee (z_{nR} < z_{3L})]
 \end{aligned}
 \tag{式35}$$

この照会で、カレントポーシヨン、S_c (すなわち、S₃) についてすべての潜在的閉塞スパンのセットを作る。照会でS₀のみ見つかる。

1 8 F) 2 4 1 2 および 2 4 1 4 F₀が空でないため、カレントポーシヨン、S_c (すなわち、S₃) を閉塞する可能性のあるスパンがなければならない。そのため、F₀の第1スパンを読み取り、新しいスパンS_Nとする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned}
 S_N &= \text{読み取り}(n) = 0; \\
 x_{NL} &= \text{読み取り}(x_L) = x_{0L}; & x_{NR} &= \text{読み取り}(x_R) = x_{0R}; \\
 z_{NL} &= \text{読み取り}(z_L) = z_{0L}; & z_{NR} &= \text{読み取り}(z_R) = z_{0R}; \\
 d_N &= \text{読み取り}(d) = d_0; & i_N &= \text{読み取り}(i) = i_0;
 \end{aligned}
 \tag{式36}$$

F₀の第1スパンの読み取りで、セットの第1 (で唯一の) スパ

ン、S₀が与えられる。読み取りプロセスではまた、(F₀₀ = 0) をセットすることでF₀からS₀を削除する。読み取りプロセスでセットF₀は空のままになる。

1 9) 2 1 0 9、2 1 0 6、2 1 1 0 および 2 1 1 2 ディスプレイスクリーンの右辺に達しておらず (すなわち、x_{CL} < x_{max}) で、現在のスパン、S_P (すなわち、S₃) も新しいスパン、S_N (すなわち、S₀) も有効である。そのため

、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：（非、非、遠、遠）。

20) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ32であることがわかる。

21) 2116、2118および2600 スパン・インタラクション・タイプがタイプ32なので、ルール5を呼び出して次のように S_3 を処理する：

21A) 2602および2606 新しいスパン、 S_N （すなわち、 S_0 ）が S_c （すなわち、 S_3 ）の x 軸投影内に隠れている。そのため、 S_N （すなわち、 S_0 ）を放棄する。 F_0 が空なので、カレントポーション、 S_c （すなわち、 S_3 ）を閉塞する可能性のあるスパンは他にない。そのため、新しいスパン、 S_N に無効とラベル付けする。

$$S_N = \text{無効} \quad (\text{式37})$$

読み取りオペレーションは終わっていない。 S_3 はまだ現在のスパン、 S_P である。セーブしたスパン、 S_s の状態は無変化のままなので、無効のままである。

22) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）

で、現在のスパンが有効で新しいスパン、 S_N は有効でないので、ルール2を呼び出して、 S_3 の処理を継続する。

22A) 2302 カレントポーション S_c （すなわち、 S_3 ）は完全に可視でなければならない。そのため、現在のスパン、 S_P （すなわち、 S_3 ）を x_{CL} から x_{CR} （すなわち、 x_{3L} から x_{3R} ）までレンダリングする。

22B) 2304および2306 セーブしたスパン、 S_s は有効でないので、 S_c のサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{CR}) \wedge (x_{nL} = x_{CR}) \\ F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{3R}) \wedge (x_{nL} = x_{3R}) \end{aligned} \quad (\text{式38})$$

この照会で、カレントポーション、 S_c （すなわち、 S_3 ）について隣接するすべてのスパンのセットを作り、これらを F_0 に保存する。照会で S_7 のみ見つかる

。

2 2 C) 2 3 0 8 および 2 3 1 0 F_0 が空でないため、隣接するスパンがなければならない。そのため、 F_0 の第 1 スパンを読み取り、現在のスパン、 S_P とする。この読み取りオペレーションで次の値をセットする：

$$S_P = \text{読み取り}(n) = 7 ;$$

$$x_{PL} = \text{読み取り}(x_L) = x_{7L} ; \quad x_{PR} = \text{読み取り}(x_R) = x_{7R} ;$$

$$x_{PL} = \text{読み取り}(z_L) = z_{7L} ; \quad z_{PR} = \text{読み取り}(z_R) = z_{7R} ; \quad (\text{式 39})$$

$$d_P = \text{読み取り}(d) = d_7 ; \quad i_P = \text{読み取り}(i) = i_7 ;$$

F_0 の第 1 スパンの読み取りで、セットの第 1 (で唯一の) スパン、 S_7 が与えられる。読み取りプロセスではまた、($F_{70} = 0$) をセットすることで F_0 から S_7 を削除する。読み取りプロセスでセット F_0 は空のままになる。

2 2 D) 2 3 1 2 カレントポーション、 S_c を次のようにセットすることで S_P (すなわち、 S_7) に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{PL} = x_{7L} ; & x_{CR} &= x_{PR} = x_{7R} ; \\ z_{CL} &= z_{PL} = z_{7L} ; & z_{CR} &= z_{PR} = z_{7R} ; \\ CF &= \max(z_{PL}, z_{PR}) = \max(z_{7L}, z_{7R}) = z_{7R} ; \end{aligned} \quad (\text{式 40})$$

これは、ステップ 2 2 C と同時に行うことができる。

2 2 E) 2 3 1 4 新しいカレントポーション、 S_c (すなわち、 S_3) があるので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各 SMCCAM ワード 9 0 0 で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \quad (\text{式 41}) \\ F_{nQ} &= F_{nV} \wedge (n \neq 7) \wedge (x_{nL} < x_{7R}) \wedge (x_{nR} > x_{7L}) \wedge [(z_{nL} < z_{7R}) \vee (z_{nR} < z_{7R})] \end{aligned}$$

この照会で、カレントポーション、 S_c (すなわち、 S_7) についてすべての潜在的閉塞スパンのセットを作る。照会でスパンは見つからず、 F_0 は空である。

2 2 F) 2 3 1 6 および 2 3 1 8 F_0 が空なので、カレントポーション、 S_c (すなわち、 S_7) を閉塞する可能性のあるスパンは他にない。そのため、新しいスパン、 S_N に無効とラベル付けする。

$$S_N = \text{無効} \quad (\text{式 42})$$

読み取りオペレーションは終わっていない。S₇はまだ現在のスパン、S_Pである。

23) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、S_Pが有効でS_Nは有効でないので、ルール2を呼び出して、S₇の処理する。

23A) 2302 カレントポーション、S_c（すなわち、S₇

）は完全に可視でなければならない。そのため、現在のスパン、S_P（すなわち、S₇）を x_{CL} から x_{CR} （すなわち x_{7L} から x_{7R} ）までレンダリングする。

23B) 2304および2306 セーブしたスパン、S_sは有効でないので、S_cのサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{CR}) \wedge (x_{AL} = x_{CR}) \\ F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{7R}) \wedge (x_{AL} = x_{7R}) \end{aligned} \quad (\text{式43})$$

この照会で、カレントポーション、S_c（すなわち、S₇）について隣接するすべてのスパンのセットを作り、これらをF_qに保存する。照会でスパンは見つからず、F_qは空である。

23C) 2308および2320 F_qが空なので、隣接するスパンはない。完全に x_{CR} （すなわち、 x_{7R} ）の左のスパンはすべて隠れていなければならないため、これらが無効と宣言する。これは、次のように各SMCCAMワード900で照会することで行う：

$$\begin{aligned} F_{AV} &= F_{AV} \wedge (x_{AR} > x_{CR}) \\ F_{AV} &= F_{AV} \wedge (x_{AR} > x_{7R}) \end{aligned} \quad (\text{式44})$$

この照会で、セットF_vのスパンを15から10に減らすことで、有効スパンの数を減らす。この時点で、F_vはS₀、S₁、S₆およびS₈からS₁₄で構成される。

23D) 2322 有効な現在のスパン、S_Pも、有効な新しいスパン、S_Nもない。故に、いずれも無効とラベルを付ける。

$$S_P = \text{無効}; \quad (\text{式45})$$

$S_N = \text{無効};$

23E) 2324 カレントポーション、 S_c を次に等しくセットする:

$$\begin{aligned} X_{CL} &= X_{CR} = X_{7R}; & X_{CR} &= \text{無効}; \\ Z_{CL} &= \text{無限}; & Z_{CR} &= \text{無効}; \\ Z_{CF} &= \text{無限} \end{aligned} \quad (\text{式46})$$

24) 2109、2106、2108および2200 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $X_{CL} < X_{max}$ ）で、現在のスパン、 S_P が有効でなく、 F_v が空でない（少なくとも一部スパンが有効であることを示す）、ルール1を呼び出す。ここで、ルール1を使って X_{7R} の最前スパンを検索する。見つかったら、この最前スパンを現在のスパン S_P として処理する。

24A) 2202 照会を行って、 X_{CL} （すなわち、 X_{7R} ）でカレントラスタラインに影響を与えそうな有効スパンすべてを見つける。照会は、次のように各SMCCAMワード900で行う:

$$\begin{aligned} F_{AQ} &= F_{AV} \wedge (x_{AL} \leq x_{CL}) \\ F_{AQ} &= F_{AV} \wedge (x_{AL} \leq x_{7R}) \end{aligned} \quad (\text{式47})$$

この照会で、 S_0 、 S_1 、 S_6 および S_8 が見つかる。照会オペレーションには、無限にセットされた Z_{CF} より小さい値をチェックする z 座標比較も含むことができる。

24B) 2204および2222 F_q が空でない（すなわち、 X_{7R} ）でカレントラスタラインに影響を与えそうな少なくとも1つのスパンがなければならない。そのため、 F_q の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする:

$$\begin{aligned} S_N &= \text{読み取り}(n) = 0; \\ X_{NL} &= \text{読み取り}(X_L) = X_{0L}; & X_{NR} &= \text{読み取り}(X_R) = X_{0R}; \\ Z_{NL} &= \text{読み取り}(Z_L) = Z_{0L}; & Z_{NR} &= \text{読み取り}(Z_R) = Z_{0R}; \\ d_N &= \text{読み取り}(d) = d_0; & i_N &= \text{読み取り}(i) = i_0; \end{aligned} \quad (\text{式48})$$

F_q の第1スパンの読み取りで、 F_q の最左スパン、 S_0 が与え

られる。読み取りプロセスではまた、 $(F_{0q} = 0)$ をセットすることで F_0 から S_0 を削除する。

2 4 C) 2 2 2 4 新しいスパン、 S_N (すなわち、 S_0) の x 座標 x_{CL} (すなわち、 x_{7R}) の z 値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{CL} - x_{NL})d_N = z_{OL} + (x_{7R} - x_{OL})d_0 \quad (式49)$$

この位置 (x_{7R} 、 z_{temp}) 2 7 0 4 を図 2 7 に示す。

2 4 D) 2 2 2 6、2 2 2 8 および 2 2 3 2 計算した値、 z_{temp} は z_{CL} (すなわち、無限) より小さいので、次のようにセットして新しいスパン、 S_N (すなわち、 S_0) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 0; \\ x_{PL} &= x_{NL} = x_{OL}; & x_{PR} &= x_{NR} = x_{OR}; \\ z_{PL} &= z_{NL} = z_{OL}; & z_{PR} &= z_{NR} = z_{OR}; \\ d_P &= d_N = d_0; & i_P &= i_N = i_0; \end{aligned} \quad (式50)$$

2 4 E) 2 2 3 6 カレントポーション、 S_c を、次のようにセットして x_{CL} (すなわち、 x_{7R}) の右の現在のスパン、 S_P (すなわち、 S_0) のポーションにセットする：

$$\begin{aligned} x_{CR} &= x_{NR} = x_{OR}; \\ z_{CL} &= z_{temp}; & z_{CR} &= z_{NR} = z_{OR}; \\ z_{CF} &= \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{OR}) = z_{temp}; \end{aligned} \quad (式51)$$

2 4 F) 2 2 3 7 および 2 2 3 8 前述の通り、この方法の本部分の目的は x_{CL} (すなわち、 x_{7R}) の最前スパンを見つけることである。 x_{CL} (すなわち、 x_{7R}) で、現在のスパン、 S_P (すなわち、 S_0) の z 座標は z_{temp} に等しい。そのため、 x_{CL} (すなわち、 x_{7R}) の S_P (すなわち、 S_0) の前にスパンがあれば、これは少なくとも部分的に z_{temp} よりオブザーバに近い位置になければなら

ない。セット F_0 は空でなく、 x_{CL} (すなわち、 x_{7R}) の最前スパンの全候補を含むので、これら候補の一部を照会で除くことができる：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{CL}) \vee (z_{nR} < z_{CL})] \\ F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{temp}) \vee (z_{nR} < z_{temp})] \end{aligned} \quad (式52)$$

この照会オペレーションで、エンドポイントの z 座標が z_{temp} より小さい候補スパンを保持する。照会后、 F_q には S_8 のみ入っている。

この方法のバリエーションとしてこの照会を省き、時間を節約することができるが、 x_{CL} (すなわち、 x_{7R}) で S_P (すなわち、 S_0) によって隠されるスパンを削除しないため、 x_{CL} (すなわち、 x_{7R}) でカレントラスタラインに影響しそうなスパンをすべて読み取らなければならない。深さ複雑性が小さいなら、このバリエーションで充分な節約ができるが、深さ複雑性が大きいと、候補スパンの除去で節約できる時間は、照会オペレーションに費やす時間を上回る。そのため、方法の選択はシーンに予測される深さ複雑性で決まる。

シーンの複雑性についてアプリオリな推測をするのを避けるため、セット F_q のスパン数を数えることで x_{CL} の深さ複雑性を決定するメカニズムを SMCCAM800 含めることができる。 F_q のスパン数が小さい場合、このステップの照会オペレーションを省略し、 F_q の各スパンを読み取る。 F_q のスパン数が充分多い場合、照会オペレーションは省略せず、 F_q の一部のスパンを削除し、充分な節約を達成する。

2 4 G) 2 2 0 4 および 2 2 2 2 F_q が空でないので、 x_{CL} (すなわち、 x_{7R}) で最前となりそうな候補スパンが少なくとももう 1 つなければならない。そのため、 F_q の第 1 スパンを読み取り、

新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_N &= \text{読み取り}(n) = 8 ; \\ x_{NL} &= \text{読み取り}(x_L) = x_{8L} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{8R} ; \\ z_{NL} &= \text{読み取り}(z_L) = z_{8L} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{8R} ; \quad (式53) \\ d_N &= \text{読み取り}(d) = d_8 ; \quad i_N = \text{読み取り}(i) = i_8 ; \end{aligned}$$

第 1 スパンの読み取りで、 F_q の唯一のスパン、 S_8 が与えられる。読み取りプロセスではまた、 $(F_{8q} = 0)$ をセットすることで F_q から S_8 を削除する。読み

取りプロセスでセット F_0 は空のままになる。

2 4 H) 2 2 2 4 新しいスパン、 S_N (すなわち、 S_8) の x 座標 x_{CL} (すなわち、 x_{7R}) の z 値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{CL} - x_{NL}) d_N = z_{8L} + (x_{7R} - x_{8L}) d_8 \quad (式 54)$$

この位置 (x_{7R} 、 z_{temp}) 2 7 0 4 を図 2 7 に示す。

2 4 1) 2 2 2 6、2 2 2 8 および 2 2 3 2 計算した値、 z_{temp} は z_{CL} (すなわち、ステップ 2 4 C で z_{temp} として計算した x_{7R} での S_0 の z 座標) より小さいので、次のようにセットして新しいスパン、 S_N (すなわち、 S_8) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 8; \\ x_{PL} &= x_{NL} = x_{8L}; & x_{PR} &= x_{NR} = x_{8R}; \\ z_{PL} &= z_{NL} = z_{8L}; & z_{PR} &= z_{NR} = z_{8R}; \\ d_P &= d_N = d_8; & l_P &= l_N = l_8; \end{aligned} \quad (式 55)$$

2 4 J) 2 2 3 6 カレントポーション、 S_c を、次のようにセットして x_{CL} (すなわち、 x_{7R}) の右の現在のスパン、 S_P (すなわち、 S_8) のポーションにセットする：

$$\begin{aligned} x_{CR} &= x_{NR} = x_{8R}; \\ z_{CL} &= z_{temp}; & z_{CR} &= z_{NR} = z_{8R}; \\ z_{CF} &= \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{8R}) = z_{8R}; \end{aligned} \quad (式 56)$$

2 4 K) 2 2 3 7、2 2 0 4、2 2 0 6 および 2 2 1 4 F_0 が空で S_P (すなわち、 S_8) が有効なので、カレントポーション、 S_c (すなわち、 x_{7R} の右の S_8) を閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各 SMCCAM ワード 9 0 0 で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 8) \wedge (x_{nL} < x_{8R}) \wedge (x_{nR} > x_{7R}) \wedge [(z_{nL} < z_{8R}) \vee (z_{nR} < z_{8R})] \end{aligned} \quad (式 57)$$

この照会で、カレントポーション、 S_c (すなわち、 x_{7R} の右の S_8) についてすべての潜在的閉塞スパンのセットを作る。照会で S_0 と S_9 が見つかる。

24L) 2216および2218 F_0 が空でないため、カレントポーション、 S_c (すなわち、 x_{7R} の右の S_8) を閉塞する可能性のあるスパンがなければならぬ。そのため、 F_0 の第1スパンを読み取り、新しいスパン S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 0 ;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{0L} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{0R} ;$$

$$z_{NL} = \text{読み取り}(z_L) = z_{0L} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{0R} ; \quad (\text{式58})$$

$$d_N = \text{読み取り}(d) = d_0 ; \quad i_N = \text{読み取り}(i) = i_0 ;$$

F_0 の第1スパンの読み取りで、セットの第1スパン、 S_0 が与えられる。読み取りプロセスではまた、($F_{0q} = 0$) をセットすることで F_0 から S_0 を削除する。

25) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$)

で、現在のスパン、 S_P (すなわち、 S_8) も新しいスパン、 S_N (すなわち、 S_0) も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：(非、非、遠、近)。

26) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ31であることがわかる。

27) 2116、2118および2500 スパン・インタラクション・タイプがタイプ31なので、ルール4を呼び出して次のように S_8 を処理する：

27A) 2502 カレントポーション、 S_c (すなわち、 x_{7R} の右の S_8) と新しいスパン、 S_N (すなわち、 S_0) がインタラクトして、 S_c をその長さのどこか： S_c (すなわち、 x_{7R} の右の S_8) と S_N (すなわち、 S_0) の間の交点で、可視から隠れに移行させる。交点を計算し、図27に示すように、(x_1 、 z_1) 2708であることがわかる。

27B) 2504 カレントポーション、 S_c (すなわち、 x_{7R} の右の S_8) を、その右エンドポイントを交点に変更して短縮する。 S_c の右エンドポイントを次のようにセットして修正する

$$\begin{aligned}
 x_{CR} &= x_i; \\
 z_{CR} &= z_i; \\
 z_{CF} &= \max(z_{CL}, z_i) = z_i;
 \end{aligned}
 \tag{式59}$$

カレントポーション、 S_c はこれで x_{7R} と x_1 の間の S_8 のポーションになる。
現在のスパン、 S_p はまだ S_8 のすべてに等しい。

2 7 C) 2 5 0 6 新しいスパン、 S_N (すなわち、 S_0) を、セーブしたスパン、 S_s としてセーブする。これは、 S_p (すなわち、 S_8) を x_1 までレンダリングする場合、後で S_s が必要とす

るために行う。

$$\begin{aligned}
 S_s &= S_N = 0; \\
 x_{SL} &= x_{NL} = x_{0L}; & x_{SR} &= x_{NR} = x_{0R}; \\
 z_{SL} &= z_{NL} = z_{0L}; & z_{SR} &= z_{NR} = z_{0R}; \\
 d_s &= d_N = d_0; & l_s &= l_N = l_0;
 \end{aligned}
 \tag{式60}$$

セーブしたスパン、 S_s の x_1 の右のポーションが x 座標 x_1 で S_c と隣接する。
方法の後半で、 S_c を x_1 までレンダリングする場合、 S_s は現在のスパン、 S_p になる。

2 7 D) 2 5 0 8 カレントポーション、 S_c (すなわち、 x_{7R} から x_1 までの S_8) が変更されたので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各 SMCCAMワード 9 0 0 で行う：

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \quad \text{(式61)} \\
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq 8) \wedge (x_{nL} < x_1) \wedge (x_{nR} > x_{7R}) \wedge [(z_{nL} < z_1) \vee (z_{nR} < z_1)]
 \end{aligned}$$

この照会で、カレントポーション、 S_c (すなわち、 x_{7R} から x_1 までの S_8) についてすべての潜在的閉塞スパンのセットを作る。照会でスパンは見つからないため、 F_0 は空である。

2 7 E) 2 5 1 0 および 2 5 1 4 F_0 が空なので、カレントポーション、 S_c (すなわち、 x_{7R} から x_1 までの S_8) を閉塞する可能性のあるスパンがあつてはならない。そのため、新しいスパン、 S_N を無効と宣言する。

$$S_N = \text{無効}; \tag{式62}$$

28) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、 S_P が有効で S_N は有効でないので、ルール2を呼び出して、 S_8 を処理する。

28A) 2302 カレントポーション、 S_c （すなわち、 x_{7R}

から x_i までの S_8 ）は完全に可視でなければならない。そのため、現在のスパン、 S_P （すなわち、 S_8 ）を x_{CL} から x_{CR} （すなわち x_{7R} から x_i ）までレンダリングする。

28B) 2304および2326 セーブしたスパン、 S_s は有効なので、 S_c のサイズを小さくする交差スパンがあった。そのため、次のようにセットして、セーブしたスパン（すなわち、 S_0 ）を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_S = 0; \\ x_{PL} &= x_{SL} = x_{OL}; & x_{PR} &= x_{SR} = x_{OR}; \\ z_{PL} &= z_{SL} = z_{OL}; & z_{PR} &= z_{SR} = z_{OR}; \\ d_P &= d_S = d_0; & i_P &= i_S = i_0; \end{aligned} \quad (\text{式63})$$

28C) 2328 カレントポーション、 S_c を、 S_P （すなわち、 S_0 ）の x_i の右のポーションにセットする。こうして、セーブしたスパン、 S_s （すなわち、 S_0 ）は x_i の左の S_8 のポーションについて隣接するスパンとして動作し、隣接は x_i で発生する。これは次のようにセットすることで行う：

$$\begin{aligned} x_{CL} &= x_{CR} = x_i; & x_{CR} &= x_{SR} = x_{OR}; \\ z_{CL} &= z_{CR} = z_i; & z_{CR} &= z_{SR} = z_{OR}; \\ z_{CF} &= \max(z_{CR}, z_{SR}) = \max(z_P, z_{OR}) = z_i; \end{aligned} \quad (\text{式64})$$

このステップは、ステップ28Bと同時に進行することができる。

28D) 2330 セーブしたスパン、 S_s の値を現在のスパン、 S_P に転送したので、 S_s を無効化する。

$$S_s = \text{無効}; \quad (\text{式65})$$

28E) 2314 新しいカレントポーション、 S_c （すなわち、 x_i の右の S_0 ）があるので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(x_{nL} < x_{CP}) \vee (x_{nR} < x_{CP})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 0) \wedge (x_{nL} < x_{0R}) \wedge (x_{nR} > x_1) \wedge [(x_{nL} < x_1) \vee (x_{nR} < x_1)] \end{aligned} \quad (\text{式66})$$

この照会で、カレントポーシオン、 S_c （すなわち、 x_1 の右の S_0 ）についてすべての潜在的閉塞スパンのセットを作る。照会で S_8 から S_{13} が見つかる。

28F) 2316および2332F₀が空でないので、カレントポーシオン S_c （すなわち、 x_1 の右の S_0 ）に影響を与えそうなスパンがなければならない。そのため、F₀の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_N &= \text{読み取り}(n) = 8; \\ x_{NL} &= \text{読み取り}(x_L) = x_{8L}; & x_{NR} &= \text{読み取り}(x_R) = x_{8R}; \\ z_{NL} &= \text{読み取り}(z_L) = z_{8L}; & z_{NR} &= \text{読み取り}(z_R) = z_{8R}; \\ d_N &= \text{読み取り}(d) = d_8; & i_N &= \text{読み取り}(i) = i_8; \end{aligned} \quad (\text{式67})$$

F₀の第1スパンの読み取りで、セットの第1スパン、 S_8 が与えられる。読み取りプロセスではまた、($F_{80} = 0$)をセットすることでF₀から S_8 を削除する。

別の方法として、交差スパンペアの左側パートナーを右側パートナーの潜在的閉塞スパンとみなされないようにする方法を含めることができる。これは、各SMCCAMワード900のnについて同時に2つの不等テストを実行できるSMCCAM800の追加照会によって行う。

29) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、現在のスパン、 S_P （すなわち、 S_0 ）も新しいスパン、 S_N （すなわち、 S_8 ）も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようにな

る：（非、近、等、非）。

30) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ21であることがわかる。

31) 2116、2118および2600 スパン・インタラクション・タイ

プがタイプ21なので、ルール5を呼び出して次のように S_0 を処理する：

31A) 2602および2604 新しいスパン、 S_N （すなわち、 S_8 ）が S_c （すなわち、 x_1 の右の S_0 ）のX軸投影内に隠れている。そのため、 S_N （すなわち、 S_8 ）を放棄する。 F_0 が空でないので、カレントポーション、 S_c （すなわち、 x_1 の右の S_0 ）を閉塞する可能性のあるスパンは他になければならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 9 ;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{9L} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{9R} ;$$

$$z_{NL} = \text{読み取り}(z_L) = z_{9L} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{9R} ; \quad (\text{式68})$$

$$d_N = \text{読み取り}(d) = d_9 ; \quad i_N = \text{読み取り}(i) = i_9 ;$$

F_0 の第1スパンの読み取りで、セットの第1スパン、 S_9 が与えられる。読み取りプロセスではまた、 $(F_{90} = 0)$ をセットすることで F_0 から S_9 を削除する。 S_0 はまだ現在のスパン、 S_P で、セーブしたスパン、 S_S の状態は無変化のままなので、無効のままである。

32) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、現在のスパン、 S_P （すなわち、 S_0 ）も新しいスパン、 S_N （すなわち、 S_9 ）も有効である。そのため、これら2つのスパン

のスパン・インタラクション・パラメータを計算し、次のようになる：（遠、遠、非、非）。

33) 2114 スパン・インタラクション・パラメータからスパン・インタラクション・タイプを決定し、タイプ5であることがわかる。

34) 2116および2400 スパン・インタラクション・タイプがタイプ5なので、ルール3を呼び出して次のように S_0 の処理を継続する：

34A) 2402 カレントポーション、 S_c （すなわち、 x_1 の右の S_0 ）が x_{NL} （すなわち、 x_{9L} ）から隠れているが、 x_{NL} （すなわち、 x_{9L} ）の左は可視である。そのため、現在のスパン、 S_P （すなわち、 S_0 ）を x_{CL} から x_{NL} （すな

わち、 x_1 から x_{9L}) までレンダリングする。

34B) 2404 現在のスパンの処理が(ここで)完了したので、次のようにセットして、新しいスパン、 S_N (すなわち、 S_3) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 9; \\ x_{PL} &= x_{NL} = x_{9L}; & x_{PR} &= x_{NR} = x_{9R}; \\ z_{PL} &= z_{NL} = z_{9L}; & z_{PR} &= z_{NR} = z_{9R}; \\ d_P &= d_N = d_9; & i_P &= i_N = i_9; \end{aligned} \quad (\text{式69})$$

34C) 2406 カレントポーション、 S_c を次のようにセットすることで S_N (すなわち、 S_9) に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{NL} = x_{9L}; & x_{CR} &= x_{NR} = x_{9R}; \\ z_{CL} &= z_{NL} = z_{9L}; & z_{CR} &= z_{NR} = z_{9R}; \\ CF &= \max(z_{NL}, z_{NR}) = \max(z_{9L}, z_{9R}) = z_{9L}; \end{aligned} \quad (\text{式70})$$

34D) 2408 セーブしたスパン、 S_s を無効化するが、これは既に無効というラベルが付いている。

$$S_s = \text{無効}; \quad (\text{式71})$$

34E) 2410 新しいカレントポーション、 S_c (すなわち、 S_9) があるので、これを閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 9) \wedge (x_{nL} < x_{9R}) \wedge (x_{nR} > x_{9L}) \wedge [(z_{nL} < z_{9L}) \vee (z_{nR} < z_{9L})] \end{aligned} \quad (\text{式72})$$

この照会で、カレントポーション、 S_c (すなわち、 S_{11}) についてすべての潜在的閉塞スパンのセットを作る。照会ではスパンは見つからず、 F_Q は空である。

34F) 2412 および 2416 F_Q が空なので、カレントポーション、 S_c (すなわち、 S_9) を閉塞する可能性のあるスパンが他にあってはならない。そのため、新しいスパン、 S_N に無効というラベルを付ける。

$$S_N = \text{無効}; \quad (\text{式73})$$

読み取りオペレーションは終わっていない。 S_9 はまだ現在のスパン、 S_P である。セーブしたスパン、 S_S の状態は無変化のままなので、無効のままである。

35) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、 S_P が有効で S_N は有効でないので、ルール2を呼び出して、 S_9 を処理する。

35A) 2302 カレントポーション、 S_c （すなわち、 S_9 ）は完全に可視でなければならない。そのため、現在のスパン、 S_P （すなわち、 S_9 ）を x_{CL} から x_{CR} （すなわち x_{9L} から x_{9R} ）までレンダリングする。

35B) 2304および2306 セーブしたスパン、 S_S は有

効でないので、 S_c のサイズを小さくする交差スパンがなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{CR}) \wedge (z_{nL} = z_{CR}) \\ F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{9R}) \wedge (z_{nL} = z_{9R}) \end{aligned} \quad (\text{式74})$$

この照会で、カレントポーション、 S_c （すなわち、 S_9 ）について隣接するすべてのスパンのセットを作り、 F_0 に保存する。照会では S_{11} のみ見つかる。

35C) 2308および2310 F_0 が空でないため、隣接するスパンがなければならない。そのため、 F_0 の第1スパンを読み取り、現在のスパン、 S_P とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_P &= \text{読み取り}(n) = 11 ; \\ x_{PL} &= \text{読み取り}(x_L) = x_{11L} ; \quad x_{PR} = \text{読み取り}(x_R) = x_{11R} ; \\ z_{PL} &= \text{読み取り}(z_L) = z_{11L} ; \quad z_{PR} = \text{読み取り}(z_R) = z_{11R} ; \\ d_P &= \text{読み取り}(d) = d_{11} ; \quad i_P = \text{読み取り}(i) = i_7 ; \end{aligned} \quad (\text{式75})$$

F_0 の第1スパンの読み取りで、セットの第1（で唯一の）スパン、 S_{11} が与えられる。読み取りプロセスではまた、 $(F_{11Q} = 0)$ をセットすることで F_0 から S_{11} を削除する。読み取りプロセスでセット F_0 は空のままになる。

35D) 2312 カレントポーション、 S_c を次のようにセットすることで S_P （すなわち、 S_{11} ）に等しくセットする：

$$\begin{aligned}
 x_{CL} &= x_{PL} = x_{11L}; & x_{CR} &= x_{PR} = x_{11R}; \\
 z_{CL} &= z_{PL} = z_{11L}; & z_{CR} &= z_{PR} = z_{11R}; \\
 CF &= \max(z_{PL}, z_{11R}) = \max(z_{11L}, z_{11R}) = z_{11R};
 \end{aligned}
 \tag{式76}$$

35E) 2314 新しいカレントポジション、 S_c (すなわち、 S_{11}) があるので、これを閉塞する可能性のあるスパンを検索す

る。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \\
 F_{nQ} &= F_{nV} \wedge (n \neq 11) \wedge (x_{nL} < x_{11R}) \wedge (x_{nR} > x_{11L}) \wedge [(z_{nL} < z_{11R}) \vee (z_{nR} < z_{11R})]
 \end{aligned}
 \tag{式77}$$

この照会で、カレントポジション、 S_c (すなわち、 S_7) についてすべての潜在的閉塞スパンのセットを作る。照会で S_0 のみ見つかる。

35F) 2316および2432 F_0 が空でないので、カレントポジション S_c (すなわち、 S_{11}) を閉塞しそうなスパンがなければならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned}
 S_N &= \text{読み取り}(n) = 0; \\
 x_{NL} &= \text{読み取り}(x_L) = x_{0L}; & x_{NR} &= \text{読み取り}(x_R) = x_{0R}; \\
 z_{NL} &= \text{読み取り}(z_L) = z_{0L}; & z_{NR} &= \text{読み取り}(z_R) = z_{0R}; \\
 d_N &= \text{読み取り}(d) = d_0; & i_N &= \text{読み取り}(i) = i_0;
 \end{aligned}
 \tag{式78}$$

F_0 の第1スパンの読み取りで、セットの第1 (で唯一の) スパン、 S_0 が与えられる。読み取り処理ではまた、($F_{00} = 0$) をセットすることで F_0 から S_0 を削除する。読み取りプロセスでセット F_0 は空のままになる。

36) 2109、2106、2110および2112 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、現在のスパン、 S_p (すなわち、 S_{11}) も新しいスパン、 S_N (すなわち、 S_0) も有効である。そのため、これら2つのスパンのスパン・インタラクション・パラメータを計算し、次のようになる：(非、非、遠、遠)。

37) 2114 スパン・インタラクション・パラメータからス

パン・インタラクション・タイプを決定し、タイプ32であることがわかる。

38) 2116、2118および2600 スパン・インタラクション・タイプがタイプ32なので、ルール5を呼び出して次のように S_{11} 进行处理する：

38A) 2602および2606 新しいスパン、 S_N （すなわち、 S_0 ）が S_c （すなわち、 S_{11} ）の x 軸投影内に隠れている。そのため、 S_N （すなわち、 S_0 ）を放棄する。 FQ が空なので、カレントポーシヨン、 S_c （すなわち、 S_{11} ）を閉塞する可能性のあるスパンは他にない。そのため、新しいスパン、 S_N に無効とラベル付けする。

$$S_N = \text{無効} \quad (\text{式79})$$

読み取りオペレーションは終わっていない。 S_s はまだ現在のスパン、 S_P である。セーブしたスパン、 S_s の状態は無変化のままなので、無効のままである。

39) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{cl} < x_{max}$ ）で、 S_P が有効で S_N は有効でないので、ルール2を呼び出して、 S_{11} の処理を継続する。

39A) 2302 カレントポーシヨン S_c （すなわち、 S_{11} ）は完全に可視でなければならない。そのため、現在のスパン、 S_P （すなわち、 S_{11} ）を x_{cl} から x_{cr} （すなわち、 x_{11L} から x_{11R} ）までレンダリングする。

39B) 2304および2306 セーブしたスパン、 S_s は有効でないので、 S_c のサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{sQ} &= F_{sV} \wedge (x_{sL} = x_{cr}) \wedge (x_{sL} = x_{cr}) \\ F_{sQ} &= F_{sV} \wedge (x_{sL} = x_{11R}) \wedge (x_{sL} = x_{11R}) \end{aligned} \quad (\text{式80})$$

この照会で、カレントポーシヨン、 S_c （すなわち、 S_{11} ）について隣接するすべてのスパンのセットを作る。照会でスパンは見つからず、 F_0 は空である。

39C) 2308および2320 F_0 が空なので、隣接するスパンはない。完全に x_{cr} （すなわち、 x_{11R} ）の左のスパンはすべて隠れていなければならないため、これらは無効と宣言する。これは、次のように各SMCCAMワード900で照会することで行う：

$$\begin{aligned} F_{\alpha V} &= F_{\alpha V} \wedge (x_{\alpha R} > x_{CR}) \\ F_{\alpha V} &= F_{\alpha V} \wedge (x_{\alpha R} > x_{11R}) \end{aligned} \quad (\text{式81})$$

この照会で、セット F_V のスパンを 10 から 5 に減らすことで、有効スパンの数を減らす。この時点で、 F_V は S_0 、 S_1 、 S_{12} から S_{14} で構成される。

39D) 2322 有効な現在のスパン、 S_P も、有効な新しいスパン、 S_N もない。故に、いずれも無効とラベルを付ける。

$$S_P = \text{無効}; \quad (\text{式82})$$

$$S_N = \text{無効};$$

39E) 2324 カレントポーション、 S_C を次に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{CR} = x_{11R}; & x_{CR} &= \text{無効}; \\ z_{CL} &= \text{無限}; & z_{CR} &= \text{無効}; \\ z_{CF} &= \text{無限} \end{aligned} \quad (\text{式83})$$

40) 2109、2106、2108 および 2200 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、現在のスパン、 S_P が有効でなく、 F_V が空でないので（少な

くとも一部スパンが有効であることを示す）、ルール 1 を呼び出す。ここで、ルール 1 を使って x_{11R} の最前スパンを検索する。見つかったら、この最前スパンを現在のスパン S_P として処理する。

40A) 2202 照会を行って、 x_{CL} （すなわち、 x_{11R} ）でカレントラスタラインに影響を与えそうな有効スパンすべてを見つける。照会は、次のように各 SMCCAM ワード 900 で行う：

$$\begin{aligned} F_{\alpha Q} &= F_{\alpha V} \wedge (x_{\alpha L} \leq x_{CL}) \\ F_{\alpha Q} &= F_{\alpha V} \wedge (x_{\alpha L} \leq x_{11R}) \end{aligned} \quad (\text{式84})$$

この照会で、 S_0 、 S_1 および S_{12} が見つかる。照会オペレーションには、無限にセットされた z_{CF} より小さい値をチェックする z 座標比較も含むことができる。

40B) 2204 および 2222 F_0 が空でないので、 x_{CL} （すなわち、 x_{11R} ）でカレントラスタラインに影響を与えそうな少なくとも 1 つのスパンがなけ

ればならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 0;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{0L}; \quad x_{NR} = \text{読み取り}(x_R) = x_{0R};$$

$$z_{NL} = \text{読み取り}(z_L) = z_{0L}; \quad z_{NR} = \text{読み取り}(z_R) = z_{0R}; \quad (\text{式85})$$

$$d_N = \text{読み取り}(d) = d_0; \quad i_N = \text{読み取り}(i) = i_0;$$

F_0 の第1スパンの読み取りで、 F_0 の最左スパン、 S_0 が与えられる。読み取りプロセスではまた、($F_{00} = 0$)をセットすることで F_0 から S_0 を削除する。

40C) 2224 新しいスパン、 S_N (すなわち、 S_0) のx座標 x_{CL} (すなわち、 x_{11R}) のz値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{CL} - x_{NL})d_N = z_{0L} + (x_{11R} - x_{0L})d_0 \quad (\text{式86})$$

この位置 (x_{11R} , z_{temp}) 2710を図27に示す。

40D) 2226、2228および2232 計算した値、 z_{temp} は z_{CL} (すなわち、無限) より小さいので、次のようにセットして新しいスパン、 S_N (すなわち、 S_0) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P = S_N = 0; \\ x_{PL} = x_{NL} = x_{0L}; \quad x_{PR} = x_{NR} = x_{0R}; \\ z_{PL} = z_{NL} = z_{0L}; \quad z_{PR} = z_{NR} = z_{0R}; \\ d_P = d_N = d_0; \quad i_P = i_N = i_0; \end{aligned} \quad (\text{式87})$$

40E) 2236 カレントポーション、 S_C を、次のようにセットして x_{CL} (すなわち、 x_{11R}) の右の現在のスパン、 S_P (すなわち、 S_0) のポーションにセットする：

$$\begin{aligned} x_{CR} = x_{NR} = x_{0R}; \\ z_{CL} = z_{temp}; \quad z_{CR} = z_{NR} = z_{0R}; \\ CF = \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{0R}) = z_{temp}; \end{aligned} \quad (\text{式88})$$

40F) 2237および2238 前述の通り、この方法の本部分の目的は x_{CL} (すなわち、 x_{11R}) の最前スパンを見つけることである。 x_{CL} (すなわち、 x_{11R}) で、現在のスパン、 S_P (すなわち、 S_0) のz座標は z_{temp} に等しい。

そのため、 x_{CL} (すなわち、 x_{IIR}) の S_P (すなわち、 S_0) の前にスパンがあれば、これは少なくとも部分的に z_{temp} よりオブザーバに近い位置になければならない。セット F_0 は空でなく、 x_{CL} (すなわち、 x_{IIR}) の最前スパンの全候補を含むので、これら候補の一部を照会で除くことができる：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{CL}) \vee (z_{nR} < z_{CL})] \\ F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{temp}) \vee (z_{nR} < z_{temp})] \end{aligned} \quad (式89)$$

この照会オペレーションで、エンドポイントの z 座標が z_{temp} よ

り小さい候補スパンを保持する。照会后、 F_0 は空である。

40G) 2204、2206および2214 F_0 が空なので、 x_{CL} (すなわち、 x_{IIR}) で最前となる候補スパンは他にあり得ない。また、 S_P (すなわち、 S_0) が有効なので、各SMCCAMワード900でスパン閉塞テスト照会を実行する：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 0) \wedge (x_{nL} < x_{OR}) \wedge (x_{nR} > x_{IIR}) \wedge [(z_{nL} < z_{temp}) \vee (z_{nR} < z_{temp})] \end{aligned} \quad (式90)$$

この照会で、カレントポーション、 S_C (すなわち、 x_{IIR} の右の S_0) についてすべての潜在的閉塞スパンのセットを作る。照会でスパンは見つからず、 F_0 は空である。

40H) 2216および2220 F_0 が空なので、カレントポーション、 S_C (すなわち、 x_{IIR} の右の S_0) を閉塞しそうなスパンは他にあり得ない。そのため、新しいスパン、 S_N に無効とラベルを付ける。

$$S_N = \text{無効}; \quad (式91)$$

読み取りオペレーションは終わっていない。 S_0 はまだ現在のスパン、 S_P である。セーブしたスパン、 S_S の状態は無変化のままなので、無効のままである。

41) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、 S_P が有効で S_N は有効でないので、ルール2を呼び出して、 S_0 を処理する。

41A) 2302 カレントポーション S_C (すなわち、 x_{IIR} の右の S_0) は完全に可視でなければならない。そのため、現在のスパン、 S_P (すなわち、 S_0)

) を x_{CL} から x_{CR} (すなわち、 x_{11L} から x_{0R}) までレンダリングする。

41B) 2304 および 2306 セーブしたスパン、 S_s は有

効でないので、 S_c のサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各 SMCCAM ワード 900 で行う：

$$\begin{aligned} F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{CR}) \wedge (z_{AL} = z_{CR}) \\ F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{OR}) \wedge (z_{AL} = z_{OR}) \end{aligned} \quad (\text{式92})$$

この照会で、カレントポーション、 S_c (すなわち、 S_o) について隣接するすべてのスパンのセットを作り、 F_o に保存する。照会でスパンは見つからず、 F_o は空である。

41C) 2308 および 2320 F_o が空なので、隣接するスパンはない。完全に x_{CR} (すなわち、 x_{OR}) の左のスパンはすべて隠れていなければならないため、これらが無効と宣言する。これは、次のように各 SMCCAM ワード 900 で照会することで行う：

$$\begin{aligned} F_{AV} &= F_{AV} \wedge (x_{AR} > x_{CR}) \\ F_{AV} &= F_{AV} \wedge (x_{AR} > x_{OR}) \end{aligned} \quad (\text{式93})$$

この照会で、セット F_v のスパンを 10 から 2 に減らすことで、有効スパンの数を減らす。この時点で、 F_v は S_1 と S_{14} で構成される。

41D) 2322 有効な現在のスパン、 S_P も、有効な新しいスパン、 S_N もない。故に、いずれも無効とラベルを付ける。

$$S_P = \text{無効}; \quad (\text{式94})$$

$$S_N = \text{無効};$$

41E) 2324 カレントポーション、 S_c を次に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{CR} = x_{OR}; & x_{CR} &= \text{無効}; \\ z_{CL} &= \text{無限}; & z_{CR} &= \text{無効}; \end{aligned} \quad (\text{式95})$$

$$z_{CF} = \text{無限}$$

42) 2109、2106、2108 および 2200 ディスブ

レイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、現在のスパン、 S_p が有効でなく、 F_v が空でないで（少なくとも一部スパンが有効であることを示す）、ルール1を呼び出す。ここで、ルール1を使って x_{OR} の最前スパンを検索する。見つかったら、この最前スパンを現在のスパン S_p として処理する。

4 2 A) 2 2 0 2 照会を行って、 x_{CL} （すなわち、 x_{OR} ）でカレントラスタラインに影響を与えそうな有効スパンすべてを見つける。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (x_{nL} \leq x_{CL}) \\ F_{nQ} &= F_{nV} \wedge (x_{nL} \leq x_{OR}) \end{aligned} \quad (\text{式96})$$

この照会で、 S_i と S_{i+1} が見つかる。照会オペレーションには、無限にセットされた z_{CF} より小さい値をチェックする z 座標比較も含むことができる。

4 2 B) 2 2 0 4および2 2 2 2 F_q が空でないで、 x_{CL} （すなわち、 x_{OR} ）でカレントラスタラインに影響を与えそうな少なくとも1つのスパンがなければならぬ。そのため、 F_q の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$\begin{aligned} S_N &= \text{読み取り}(n) = 1 ; \\ x_{NL} &= \text{読み取り}(x_L) = x_{iL} ; \quad x_{NR} = \text{読み取り}(x_R) = x_{iR} ; \\ z_{NL} &= \text{読み取り}(z_L) = z_{iL} ; \quad z_{NR} = \text{読み取り}(z_R) = z_{iR} ; \quad (\text{式97}) \\ d_N &= \text{読み取り}(d) = d_i ; \quad i_N = \text{読み取り}(i) = i_i ; \end{aligned}$$

第1スパンの読み取りで F_q の最左スパン、 S_i が与えられる。読み取りプロセスではまた、 $(F_{iq} = 0)$ をセットすることで F_q から S_i を削除する。

4 2 C) 2 2 2 4 新しいスパン、 S_N （すなわち、 S_i ）の x 座標 x_{CL} （すなわち、 x_{OR} ）の z 値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{CL} - x_{NL})d_N = z_{iL} + (x_{OR} - x_{iL})d_i \quad (\text{式98})$$

この位置（ x_{OR} 、 z_{temp} ）2 7 1 2を図27に示す。

4 2 D) 2 2 2 6、2 2 2 8および2 2 3 2 計算した値、 z_{temp} は z_{CL} （すなわち、無限）より小さいので、次のようにセットして新しいスパン、 S_N （す

なわち、 S_1) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 1; \\ x_{PL} &= x_{NL} = x_{1L}; & x_{PR} &= x_{NR} = x_{1R}; \\ z_{PL} &= z_{NL} = z_{1L}; & z_{PR} &= z_{NR} = z_{1R}; \\ d_P &= d_N = d_1; & i_P &= i_N = i_1; \end{aligned} \quad (式 99)$$

4 2 E) 2 2 3 6 カレントポーション、 S_c を、次のようにセットして x_{CL} (すなわち、 x_{OR}) の右の現在のスパン、 S_P (すなわち、 S_1) のポーションにセットする：

$$\begin{aligned} x_{CR} &= x_{NR} = x_{1R}; \\ z_{CL} &= z_{temp}; & z_{CR} &= z_{NR} = z_{1R}; \\ CF &= \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{1R}) = z_{temp}; \end{aligned} \quad (式 100)$$

4 2 F) 2 2 3 7 および 2 2 3 8 前述の通り、この方法の本部分の目的は x_{CL} (すなわち、 x_{OR}) の最前スパンを見つけることである。 x_{CL} (すなわち、 x_{OR}) で、現在のスパン、 S_P (すなわち、 S_1) の z 座標は z_{temp} に等しい。そのため、 x_{CL} (すなわち、 x_{OR}) の S_P (すなわち、 S_1) の前にスパンがあれば、これは少なくとも部分的に z_{temp} よりオブザーバに近い位置になければならない。セット F_0 は空でなく、 x_{CL} (すなわち、 x_{OR}) の最前スパンの全候補を含むので、これら候補の一部を照会で除くことができる：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{CL}) \vee (z_{nR} < z_{CL})] \\ F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge [(z_{nL} < z_{temp}) \vee (z_{nR} < z_{temp})] \end{aligned} \quad (式 101)$$

この照会オペレーションで、エンドポイントの z 座標が z_{temp} より小さい候補スパンを保持する。照会后、 F_0 には S_{14} しか入っておらず、照会によって変更されていない。

4 2 G) 2 2 0 4 および 2 2 2 2 F_0 が空でないので、 x_{CL} (すなわち、 x_{OR}) で最前となりそうなスパンが少なくとももう 1 つなければならない。そのため、 F_0 の第 1 スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 14;$$

$x_{NL} = \text{読み取り}(x_L) = x_{14L}; \quad x_{NR} = \text{読み取り}(x_R) = x_{14R};$
 $z_{NL} = \text{読み取り}(z_L) = z_{14L}; \quad z_{NR} = \text{読み取り}(z_R) = z_{14R}; \quad (\text{式 } 102)$
 $d_N = \text{読み取り}(d) = d_{14}; \quad i_N = \text{読み取り}(i) = i_{14};$

第1スパンの読み取りで、 F_0 の唯一のスパン、 S_{14} が与えられる。読み取りプロセスではまた、($F_{140} = 0$)をセットすることで F_0 から S_{14} を削除する。読み取りプロセスでセット F_0 は空のままになる。

4 2 H) 2 2 2 4 新しいスパン、 S_N (すなわち、 S_{14}) の x 座標 x_{CL} (すなわち、 x_{0R}) の z 値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{CL} - x_{NL})d_N = z_{14L} + (x_{0R} - x_{14L})d_{14} \quad (\text{式 } 103)$$

この位置 (x_{0R} 、 z_{temp}) 2 7 1 4 を図 2 7 に示す。

4 2 1) 2 2 2 6、2 2 2 8 および 2 2 3 2 計算した値、 z_{temp} は、 z_{CL} (すなわち、ステップ 4 2 C) で z_{temp} として計算した x_{7R} での S_0 の z 座標) より小さいので、次のようにセットして新しいスパン、 S_N (すなわち、 S_{14}) を現在のスパン、 S_P にする

:

$$\begin{aligned}
 S_P &= S_N = 14; \\
 x_{PL} &= x_{NL} = x_{14L}; & x_{PR} &= x_{NR} = x_{14R}; \\
 z_{PL} &= z_{NL} = z_{14L}; & z_{PR} &= z_{NR} = z_{14R}; \\
 d_P &= d_N = d_{14}; & i_P &= i_N = i_{14};
 \end{aligned} \quad (\text{式 } 104)$$

4 2 J) 2 2 3 6 カレントポーション、 S_c を、次のようにセットして x_{CL} (すなわち、 x_{0R}) の右の現在のスパン、 S_P (すなわち、 S_{14}) のポーションにセットする:

$$\begin{aligned}
 x_{CR} &= x_{NR} = x_{14R}; \\
 z_{CL} &= z_{temp}; & z_{CR} &= z_{NR} = z_{14R}; \\
 CP &= \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{14R}) = z_{14R};
 \end{aligned} \quad (\text{式 } 105)$$

図 2 7 に示すように、 z_{temp} は z_{CL} と等しいので、 $\max()$ ファンクションで z_{temp} または z_{CL} を選ぶことができる。

42K) 2237、2204、2206および2214 F_q が空で S_p （すなわち、 S_{14} ）が有効なので、カレントポーション、 S_c （すなわち、 x_{0R} の右の S_{14} ）を閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_p) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CP}) \vee (z_{nR} < z_{CP})] \quad (\text{式 } 106) \\ F_{nQ} &= F_{nV} \wedge (n \neq 8) \wedge (x_{nL} < x_{14R}) \wedge (x_{nR} > z_{temp}) \wedge [(z_{nL} < z_{14R}) \vee (z_{nR} < z_{14R})] \end{aligned}$$

この照会で、カレントポーション、 S_c （すなわち、 x_{0R} の右の S_{14} ）についてすべての潜在的閉塞スパンのセットを作る。照会ではスパンが見つからず、 F_q は空である。

42L) 2216および2220 F_q が空なので、カレントポーション、 S_c （すなわち、 x_{0R} の右の S_{14} ）を閉塞しそうなスパンは他にあり得ない。そのため、新しいスパン、 S_N に無効とラベルを付ける。

$$S_N = \text{無効}; \quad (\text{式 } 107)$$

読み取りオペレーションは終わっていない。 S_{14} はまだ現在のスパン、 S_p である。セーブしたスパン、 S_s の状態は無変化のままなので、無効のままである。

43) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、 S_p が有効で S_N は有効でないので、ルール2を呼び出して、 S_{14} を処理する。

43A) 2302 カレントポーション S_c （すなわち、 x_{0R} の右の S_{14} ）は完全に可視でなければならない。そのため、現在のスパン、 S_p （すなわち、 S_{14} ）を x_{CL} から x_{CR} （すなわち、 x_{0R} から x_{14R} ）までレンダリングする。

43B) 2304および2306 セーブしたスパン、 S_s は有効でないので、 S_c のサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{CR}) \wedge (z_{nL} = z_{CR}) \quad (\text{式 } 108) \\ F_{nQ} &= F_{nV} \wedge (x_{nL} = x_{14R}) \wedge (z_{nL} = z_{14R}) \end{aligned}$$

この照会で、カレントポーション、 S_c （すなわち、 x_{0R} の右の S_{14} ）につい

て隣接するすべてのスパンのセットを作り、 F_0 に保存する。照会でスパンは見つからず、 F_0 は空である。

4 3 C) 2 3 0 8 および 2 3 2 0 F_0 が空なので、隣接するスパンはない。完全に x_{CR} (すなわち、 x_{14R}) の左のスパンはすべて隠れていなければならないため、これらが無効と宣言する。これは、次のように各 SMCCAMワード 9 0 0 で照会することで行う：

$$\begin{aligned} F_{AV} &= F_{AV} \wedge (x_{AR} > x_{CR}) \\ F_{AV} &= F_{AV} \wedge (x_{AR} > x_{14R}) \end{aligned} \quad (\text{式 109})$$

この照会で、セット F_V のスパンを 2 から 1 に減らすことで、有効スパンの数を減らす。この時点で、 F_V は S_1 のみから構成される。

4 3 D) 2 3 2 2 有効な現在のスパン、 S_P も、有効な新しいスパン、 S_N もない。故に、いずれも無効とラベルを付ける。

$$S_P = \text{無効}; \quad (\text{式 110})$$

$$S_N = \text{無効};$$

4 3 E) 2 3 2 4 カレントポーション、 S_C を次に等しくセットする：

$$\begin{aligned} x_{CL} &= x_{CR} = x_{14R}; & x_{CR} &= \text{無効}; \\ z_{CL} &= \text{無限}; & z_{CR} &= \text{無効}; \\ z_{CF} &= \text{無限} \end{aligned} \quad (\text{式 111})$$

4 4) 2 1 0 9、2 1 0 6、2 1 0 8 および 2 2 0 0 ディスプレイスクリーンの右辺に達しておらず (すなわち、 $x_{CL} < x_{max}$) で、現在のスパン、 S_P が有効でなく、 F_V が空でない (少なくとも一部スパンが有効であることを示す)、ルール 1 を呼び出す。ここで、ルール 1 を使って x_{14R} の最前スパンを検索する。見つかったら、この最前スパンを現在のスパン S_P として処理する。

4 4 A) 2 2 0 2 照会を行って、 x_{CL} (すなわち、 x_{14R}) でカレントラスタラインに影響を与えそうな有効スパンすべてを見つける。照会は、次のように各 SMCCAMワード 9 0 0 で行う：

$$\begin{aligned} F_{AQ} &= F_{AV} \wedge (x_{AL} \leq x_{CL}) \\ F_{AQ} &= F_{AV} \wedge (x_{AL} \leq x_{14R}) \end{aligned} \quad (\text{式 112})$$

この照会で、 S_1 のみ見つかる。照会オペレーションには、無限にセットされた z_{cf} より小さい値をチェックする z 座標比較も含む

ことができる。

4 4 B) 2 2 0 4 および 2 2 2 2 F_0 が空でないので、 x_{cl} (すなわち、 x_{14R}) でカレントラスタラインに影響を与えそうな少なくとも1つのスパンがなければならない。そのため、 F_0 の第1スパンを読み取り、新しいスパン、 S_N とする。この読み取りオペレーションで次の値をセットする：

$$S_N = \text{読み取り}(n) = 1;$$

$$x_{NL} = \text{読み取り}(x_L) = x_{1L}; \quad x_{NR} = \text{読み取り}(x_R) = x_{1R};$$

$$z_{NL} = \text{読み取り}(z_L) = z_{1L}; \quad z_{NR} = \text{読み取り}(z_R) = z_{1R}; \quad (\text{式 113})$$

$$d_N = \text{読み取り}(d) = d_1; \quad i_N = \text{読み取り}(i) = i_1;$$

第1スパンの読み取りで、 F_0 の最左スパン、 S_1 が与えられる。読み取り処理ではまた、($F_{10} = 0$) をセットすることで F_0 から S_1 を削除する。

4 4 C) 2 2 2 4 新しいスパン、 S_N (すなわち、 S_1) の x 座標 x_{cl} (すなわち、 x_{14R}) の z 値を z_{temp} として計算する。

$$z_{temp} = z_{NL} + (x_{cl} - x_{NL}) d_N = z_{1L} + (x_{14R} - x_{1L}) d_1 \quad (\text{式 114})$$

この位置 (x_{14R} 、 z_{temp}) 2 7 1 6 を図 2 7 に示す。

4 4 D) 2 2 2 6、2 2 2 8 および 2 2 3 2 計算した値、 z_{temp} は z_{cl} (すなわち、無限) より小さいので、次のようにセットして新しいスパン、 S_N (すなわち、 S_1) を現在のスパン、 S_P にする：

$$\begin{aligned} S_P &= S_N = 1; \\ x_{PL} &= x_{NL} = x_{1L}; & x_{PR} &= x_{NR} = x_{1R}; \\ z_{PL} &= z_{NL} = z_{1L}; & z_{PR} &= z_{NR} = z_{1R}; \\ d_P &= d_N = d_1; & i_P &= i_N = i_1; \end{aligned} \quad (\text{式 115})$$

4 4 E) 2 2 3 6 カレントポーション、 S_c を、次のようにセ

ットして x_{cl} (すなわち、 x_{14R}) の右の現在のスパン、 S_P (すなわち、 S_1)

のポーションにセットする：

$$\begin{aligned} x_{CR} &= x_{NR} = x_{1R}; \\ z_{CL} &= z_{temp}; \quad z_{CR} = z_{NR} = z_{1R}; \\ CF &= \max(z_{temp}, z_{NR}) = \max(z_{temp}, z_{1R}) = z_{temp}; \end{aligned} \quad (\text{式 116})$$

44F) 2237、2204、2206および2214 F_Q が空で S_P （すなわち、 S_1 ）が有効なので、カレントポーション、 S_C （すなわち、 x_{14R} の右の S_1 ）を閉塞する可能性のあるスパンを検索する。スパン閉塞テスト照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{nQ} &= F_{nV} \wedge (n \neq S_P) \wedge (x_{nL} < x_{CR}) \wedge (x_{nR} > x_{CL}) \wedge [(z_{nL} < z_{CF}) \vee (z_{nR} < z_{CF})] \\ F_{nQ} &= F_{nV} \wedge (n \neq 8) \wedge (x_{nL} < x_{1R}) \wedge (x_{nR} > x_{14R}) \wedge [(z_{nL} < z_{temp}) \vee (z_{nR} < z_{temp})] \end{aligned} \quad (\text{式 117})$$

この照会で、カレントポーション、 S_C （すなわち、 x_{14R} の右の S_1 ）についてすべての潜在的閉塞スパンのセットを作る。照会ではスパンが見つからず、 F_Q は空である。

44G) 2216および2220 F_Q が空なので、カレントポーション、 S_C （すなわち、 x_{14R} の右の S_1 ）を閉塞しそうなスパンは他にあり得ない。そのため、新しいスパン、 S_N に無効とラベルを付ける。

$$S_N = \text{無効}; \quad (\text{式 118})$$

読み取りオペレーションは終わっていない。 S_1 はまだ現在のスパン、 S_P である。セーブしたスパン、 S_S の状態は無変化のままなので、無効のままである。

45) 2109、2106、2110および2300 ディスプレイスクリーンの右辺に達しておらず（すなわち、 $x_{CL} < x_{max}$ ）で、 S_P が有効で S_N は有効でないので、ルール2を呼び出して、 S_1 を処理する。

45A) 2302 カレントポーション S_C （すなわち、 x_{14R} の右の S_1 ）は完全に可視でなければならない。そのため、現在のスパン、 S_P （すなわち、 S_1 ）を x_{CL} から x_{CR} （すなわち、 x_{14R} から x_{1R} ）までレンダリングする。

45B) 2304および2306 セーブしたスパン、 S_S は有効でないので、 S_C のサイズを小さくする交差スパンはなかった。そのため、照会を行って隣接スパンを探す。照会は、次のように各SMCCAMワード900で行う：

$$\begin{aligned} F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{CR}) \wedge (x_{AL} = x_{CR}) \\ F_{AQ} &= F_{AV} \wedge (x_{AL} = x_{IR}) \wedge (x_{AL} = x_{IR}) \end{aligned} \quad (\text{式 } 119)$$

この照会で、カレントポーション、 S_C （すなわち、 x_{14R} の右の S_I ）について隣接するすべてのスパンのセットを作り、 F_Q に保存する。照会でスパンは見つからず、 F_Q は空である。

45C) 2308および2320 F_Q が空なので、隣接するスパンはない。完全に x_{CR} （すなわち、 x_{14R} ）の左のスパンはすべて隠れていなければならないため、これらが無効と宣言する。これは、次のように各SMCCAMワード900で照会することで行う：

$$\begin{aligned} F_{AV} &= F_{AV} \wedge (x_{AR} > x_{CR}) \\ F_{AV} &= F_{AV} \wedge (x_{AR} > x_{IR}) \end{aligned} \quad (\text{式 } 120)$$

この照会で、セット F_V のスパンを1から0に減らすことで、有効スパンの数を減らす。この時点で、 F_V は空である。

45D) 2322 有効な現在のスパン、 S_P も、有効な新しいスパン、 S_N もない。故に、いずれも無効とラベルを付ける。

$$\begin{aligned} S_P &= \text{無効}; \\ S_N &= \text{無効}; \end{aligned} \quad (\text{式 } 121)$$

45E) 2324 カレントポーション、 S_C を次に等しくセッ

トする：

$$\begin{aligned} x_{CL} &= x_{CR} = x_{IR}; & x_{CR} &= \text{無効}; \\ z_{CL} &= \text{無限}; & z_{CR} &= \text{無効}; \\ z_{CF} &= \text{無限} \end{aligned} \quad (\text{式 } 122)$$

46) 2106、2108および2120 現在のスパン、 S_P が有効でなく、 F_V は空で、すべてのスパンを処理した。 x_{CL} （すなわち、 x_{IR} ）は x_{max} より小さくならないので、スパンのカバーしている x_{CL} （すなわち、 x_{IR} ）の右にまだカレントラスタラインの一部がある。そのため、背景を x_{CL} （すなわち、 x_{IR} ）から x_{max} までレンダリングする。カレントラスタラインについてセグメント・スパン・ラスタリゼーション方法2100は完了する。

スパン・ソーティング・パイプライン400内の処理のタイミング

ジェネリック3Dグラフィックス・パイプライン200は全体として2ページのメモリを持つダブルバッファされたフレームバッファ224を用いる。これによって、1ページの画像をディスプレイ装置に表示しながら、レンダリングプロセスでもう1枚のページに画像を書き込むことができる。幾何形状が空間ソーティングなしにパイプラインに送られるため、画像データはディスプレイスクリーン104内のランダムな位置に書き込まれる。シーンレンダリングが完了すると、ページがスワップされ、次のシーンがレンダリングされる。

ディスプレイメモリの数個のラスタラインのみ使用

スパン・ソーティング・レンダリング・パイプライン400は、1つのラスタラインについて完全なピクセルカラー化を生成してから、次のラスタラインに進む。ラスタラインのレンダリングを表示に遅れない速度で行う場合、ラスタラインメモリ416に数個のラ

スタラインしか必要としない。

図28に、ラスタラインメモリ416のラスタライン2個のみで行えるスパン・ソーティング・レンダリング・パイプライン400内の処理のタイミングを示す。この図において、水平軸は時間で、目盛りマークがそれぞれ物理的ディスプレイ装置(CRT等)に1ラスタラインを表示する時間を表す。シーン1 2802の多角形処理1400は、シーンの多角形をすべてバケット・ソート・メモリ404に書き込む。この2802を完了すると、シーン1についてラスタライン0がレンダリングされる2804。この第1ラスタラインのレンダリング2804が完了すると、後続ラスタラインをディスプレイ装置が必要とする前にレンダリングした場合、ディスプレイ装置へのシーン1の表示2806を開始できる。このように、ラスタラインのレンダリング2808は、ラスタラインの表示2806と共にロックステップにあり、ラスタラインを処理しながら書き込む1つと、読み取ってディスプレイ装置にデータを送る1つという、ラスタラインメモリ416のラスタライン2個しか必要としない。シーン1 2802の多角形処理1400が完了すると、シーン2 2810の多角形処理1400を開始でき

、この2810を、シーン1 2808のラスタライン処理と、シーン1 2806のラスタライン表示と平行して（パイプライン方式で）行う。

図28の図は、ラスタライン処理2808がラスタラインの表示にかかる時間に非常に近接して一致すると想定しているため、理想的なケースである。実際のシステムでは、ラスタラインの処理にかかる時間にはややバリエーションがある。そのため、いくつかのラスタラインは、対応するシーンを表示する前に処理することによって、タイミング要件に「スラック」を与えなければならない。図28の図は水平や縦リトレース時間などの表示タイミング要件も無視

しているが、これらはタイミングにある程度の「スロップ」を導入するために用いることができる。

ピクセルカラー化処理が、ラスタラインのすべての多角形スパンのピクセル数ではなくディスプレイスクリーンのピクセル数に比例するため、フルフレームバッファよりメモリを少なくすることができる。レンダリングのタイミングをこのようにすると、スパン・ソーティング・レンダリング・パイプライン400の待ち時間は大半が多角形処理1400部分となる。多角形処理1400がディスプレイスクリーン速度に遅れなければ、ディスプレイ装置のスクリーンをリフレッシュする毎に新しいシーンを表示することができる。

図28の図は、新しいシーンを表示する時、シーンのラスタライン0を先に表示しなければならないと想定している。この想定により、多角形処理2802とラスタライン処理2808がディスプレイ装置2806に「シンクアップ」しなければならない。別のアプローチとして、多角形処理をディスプレイ装置のサイクル2806内のあらゆるポイントで完了させ、表示するラスタラインでラスタラインレンダリングを開始（ラスタライン処理の待ち時間を考慮して）するものがある。この代替えアプローチには、バケット・ソート・メモリ404がどのラスタラインでも読み取りオペレーションを開始できなければならない。

標準型テレビなど一部のディスプレイスクリーンは、ラスタラインのインターレースディスプレイを用いている。これは、偶数番号のラスタラインを先に（偶数「フィールド」）、奇数番号のラスタラインを次に（奇数「フィールド」）表

示するものである。この種のディスプレイでは、スパン・ソーティング・レンダリング・パイプライン400はディスプレイが必要とする順番でラスタラインを生成する。ディスプレイの2つのフィールドは別のシーンとして、

あるいは1つのインターレースシーンとして扱うことができる。

シングルバッファのディスプレイメモリの使用

図28では、ラスタラインメモリ416には数個のラスタラインしか保存しないと想定している。別のアプローチとして、ラスタラインメモリ416にラスタラインすべてについてメモリを持たせるものがある。これによって必要なメモリが非常に大きく（おそらく100倍以上）なるが、大半のコンピュータのビデオディスプレイシステムはスクリーンの各ピクセルについてメモリを持っており、3Dレンダリングはこの種のコンピュータディスプレイのアドオンと考えることができる。そのため、必要なメモリは既にホストコンピュータシステムのビデオメモリ中に存在している場合がある。

本書の本部分では、ラスタラインメモリ416は各ラスタラインについてすべてのピクセルカラー情報を保存すると想定する。図29に、10個のラスタライン（実用的なディスプレイは一般に1000のオーダーのラスタラインを有する）を連続してリフレッシュするディスプレイ装置のタイミングを示すが、ここではラスタラインに相当する3枚のスクリーンが表示される2902。前の図のように、水平軸が時間で、各目盛りマークが物理的ディスプレイ装置に1つのラスタラインを表示する時間を表す。シーン表示中のあるポイント2904で、次のシーンの多角形処理1400を完了し2906、このポイント2906はいかなる方法でもシーン表示にシンクロするとは想定されない。多角形処理1400の完了2906は、ラスタライン処理2907が開始できることを意味する。ラスタライン0の処理2908を先に行い、このラスタライン0をディスプレイ装置に送る次回2910にラスタラインを表示する。同様に、ラスタラインを処理し2912、その後表示し2914、もってシーン全体を表示する2916。図29は、シーンの多角形処理

1400の完了時2906、最後のラスタラインのラスタライン処理2912が、最初のラスタライン表示2910後に起こっても、ディスプレイのすぐ次のリフレッシュサイクル2916からシーンを表示する方法を示す。多角形処理とシーン全体のラスタライン処理の両方がディスプレイ装置のスクリーンのリフレッシュレートに遅れなければ、新しいシーンをスクリーンリフレッシュ毎に表示することができる。場合によって多角形処理またはラスタライン処理のいずれかが追いつかなかった場合、ディスプレイ装置は2つの連続するディスプレイリフレッシュサイクルで同じシーンを表示することができる。

図29の背後にあるコンセプトを、平均してディスプレイリフレッシュサイクルに追いつかないラスタライン処理2912に拡大することができる。すなわち、最後のラスタラインのラスタライン処理2912が表示2914の前に完了する限り、シーン全体が正しく表示される。さらに、ラスタライン処理は、最も最近表示されたばかりのラスタラインについて開始することができる。図30は、ラスタライン7が完全に表示される直前に多角形処理1400が完了3002するところを示す。ラスタライン処理3004は、ラスタライン7 3006で始まることができ、このラスタラインは、ディスプレイ装置がそれを表示する次回に表示される3008。シーンの表示は、最後に処理されたラスタライン3010がその表示3012前に完了している限り、前のシーンの表示と混合しない（すなわちクリーンなトランジション）。このように、すべてのラスタを処理する最大時間3014はほぼ2つのスクリーンリフレッシュサイクルと同じ長さである。故に、各ラスタラインの処理は、ダブルバッファを必要とすることなく、シングルバッファのみで、表示する時間の約2倍の長さとなる。

図30に示すコンセプトは、ラスタライン処理3004が1以上のスクリーンリフレッシュサイクルをカバーするものだが、これをラスタライン処理3004がラスタライン0の表示まで待つようにして、リフレッシュサイクルと同期するよう修正することができる。次のシーンの多角形処理1400は、バケット・ソート・メモリ404がダブルバッファであるため、待つ必要はない。

ダブルバッファのディスプレイメモリの使用

スパン・ソーティング・レンダリング・パイプライン400は、完全ダブルバッファのラスタラインメモリ416と共に機能することができ、ここでディスプレイスクリーン104の各ピクセルはメモリの2ワード（各バッファページに1つずつ）に対応する。この構成では、あるシーンの多角形処理104は完了するまで実行し、そして、ディスプレイスクリーン104のすべてのラスタラインが処理されるまでラスタライン処理を実行し、そしてシーンをディスプレイ装置に表示する。このアプローチには、ディスプレイ装置のタイミングに制約されずに必要なだけ長く時間をかけられるという利点がある。このアプローチには、追加待ち時間段階を導入するという不都合がある。

照会プロセッサのアーキテクチャ

図5の各ラスタラインに関するスパン・ソーティング・レンダリング・アーキテクチャ500内では、照会プロセッサ510はスパン生成プロセッサ508からスパンを受け取り、可視スパンポーション（前述）を決定し、その可視スパンポーションをラスタライズ・プロセッサ510に送る。照会プロセッサ510のブロック図を、スパンメモリ408内の2ページと共に図31に示す。

一方のスパンメモリページ800はスパン生成プロセッサ508からスパンパラメータを受け取り、他方のページ800は照会され

、可視スパンポーションを決定する。2つのページ800を、マルチプレクサのセットである照会プロセッサクロスバー3102を介してスワップする。照会プロセッサ510には、スパンパラメータを保存する次の5組のレジスタに細分されたレジスタファイル3104が含まれる：1) カレントポーションの S_c のレジスタ3106、2) 現在のスパン、 S_p のレジスタ3108、3) 新しいスパン S_n のレジスタ3110、4) セーブしたスパン、 S_s のレジスタ3112、および5) スパン交差位置などのために使うテンポラリレジスタ3114。

照会プロセッサコントロール3116は、照会オペレーションのシーケンスを含む照会プロセッサ510のオペレーションのシーケンスを決定する。ラスタライン処理が進むにつれ、方法1800または2100内のブランチは、照会結果、スパン状態およびスパン・インタラクション・タイプ（SIT）で決まる。S

cおよび S_N の各ペアにつき、SITおよびSIP論理3118がスパン・インタラクション・タイプを生成する。SITの計算には、式10の演算と、各種比較演算を実行する必要がある。これら演算は、レジスタファイル3104から受け取ったデータについて、比較およびオフセット計算3200ブロックで行う。ルール4 2500を呼び出す時は常に、交差計算ブロックで S_c と S_N の交点を計算する。

スループットを最大にするため、比較およびオフセット計算3200を並列ハードウェアにインプリメントすることができる。図32に4個の同一ブロック3202を示すが、各ブロック3202には2つの算術コンパレータ3204および3206、減算器3208、乗算器3210を含む。減算器3208と乗算器3210は式10に示すオフセットを計算する。コンパレータは、スパン・インタラクション・パラメータ(SIP)の生成に用いることのできる

信号を生成する。ブロック3202への入力、レジスタファイル3104から入り、出力はSITおよびSIP論理3118へ出る。

ソーティング・マグニチュード比較CAM回路

図33に、CMOSトランジスタとゲートから構成されるDuluk 特許による先行技術のMCCAMビットセル3300を示す。MCCAMワード内のフィールドは、多数のこれらセル3300で構築できる。このセル3300で、1ビットは6トランジスタRAMサブ回路3302に保存され、読み取りと書き込みは信号CamWord 3304で制御し、データは差分ビット線、BitH3306およびBitL3308を介して読み取り・書き込みされる(信号名末尾の「H」および「L」は、それぞれアクティブハイとアクティブローを示す)。照会オペレーションにおいて、XORサブ回路3310は排他的OR関数を実行して信号NotEqual 3312を出す、これはRAM3302に保存されるビット値がビット線3306および3308のビット値と等しくないことを示す。照会オペレーション中、表明される信号EnableIn 3314は、フィールドの上位ビットがすべて等しいことを示し、このビット位置は比較に使用できる。EnableIn 3314を表明した時NotEqual 3312が論理「0」だと、予めチャージしたイネーブル・プロパゲーション

オン・サブ回路3316が、EnableOut 3318を表明することで次に下位のビット位置を使用可能にする。EnableIn 3314を表明した時NotEqual 3312が論理「1」だと、2トランジスタ照会解像サブ回路3320が、ComparisonOp 3324 (Duluk 特許記述の通り) の値に応じて条件付きでQueryTrue 3322信号をプルダウンし、EnableOut 3318は表明されないままとなる。

MCCAMとは対照的に、SMCCAM800 (「S」は「ソー

ティング」を表すことに注意) は、ソーティングオペレーション中、1つのワード900から次へデータをシフトできる。図34は、SMCCAMワード900中の1ビットの「TTLタイプ」回路である。このセル3400にはビットを保存するためのマスタスレーブ・フリップフロップ3402を含むが、これがシフトレジスタでビットとして機能しなければならないためである。このセルを、別個のI/Oライン、InputBit 3404とOutputBit 3406と共に示すが、1つの双方向I/Oラインまたは双方向差分ペアを用いることもできる。読み取りオペレーションは、Read 3408制御ラインを表明することで行い、これがトライステートバッファ3410を使用可能にする。書き込みオペレーションは、WriteEn 3414が表明されShiftEn 3416が表明されない時、Clock 3412の立ち上がりエッジで行われる。ソートオペレーション内のシフトは、WriteEn 3414が表明されShiftEn 3416も表明されている時、Clock 3412の立ち上がりエッジで行われる。信号ShiftEn 3416は、InputBit 3404とShiftIn 3420のデータのいずれをフリップフロップ3402に保存するかを選択するマルチプレクサ3418を制御する。ShiftIn 3420のデータは、前のSMCCAMワード900の対応するセル3400のShiftOut 3422から来る。照会オペレーションは、Duluk 特許のMCCAMと同様に機能する。すなわち、排他的OR 3424で信号NotEqual 3312を生成し、これを使ってEnableOut 3318とワードの照会結果StoredGreaterL 3426を生成する。照会オペレーション機能の大きな違いは、1) セルにComparisonOp 3324がないこと、および2) 照会結果を生成するためのオープンコレクタ回路3428である。

図35は、SMCCAMワード900の1ビットのための27ト

ランジスタCMOS回路である。このセル3500は、フリップフロップのマスタ部分としての6トランジスタRAMサブ回路3302と、フリップフロップのスレーブ部分としてのスレーブ・メモリ・サブ回路3502を含む。6トランジスタRAMサブ回路3302は従来の方法で読み取りおよび書き込みオペレーションに使い、スレーブ・メモリ・サブ回路3502はシフト中にのみ用いる。ビットを6トランジスタRAMサブ回路3302にシフトできるようにするため、セル3504にシフト・ライト・サブ回路を含む。シフトを行うと（この例のセル3500内で）次のシーケンスが発生する：1）差分クロック信号ShiftToSlaveH 3506とShiftToSlaveL 3508を表明してから、デアサーションして、データビットをスレーブ・メモリ・サブ回路3502にラッチする、2）データビットを、差分信号ShiftDataOutH 3510とShiftDataOutL 3512を介して、セルから次のワード900の対応するセル3500に出力する、そして3）データビットを次のワード900の差分信号ShiftDataInH 3514とShiftDataInL 3516で受け取って、ShiftEnable 3518を表明することでそのセル3500に書き込む。

多くのビットセル回路を等価関数を行うよう設計することができる。その他の選択例を示すため、図35のセルは予めチャージされていないイネーブル・プロパゲーション・サブ回路3520と3トランジスタ照会解像サブ回路3522を使っている。スレーブ・メモリ・サブ回路3502はダイナミック回路で、静的ラッチを含む他の様々なタイプのラッチと置き換えられることに注意。

図36は、SMCCAMワード900の1ビットの23トランジスタCMOS回路である。このセル3600は、フリップフロップのマスタハーフとスレーブハーフの両方で動的記憶装置を用いる1

2トランジスタ・マスタ・スレーブ・メモリ・サブ回路3602を使っている。このセルの大きな利点は、トランジスタの数が少なく、直列トランジスタ（直列トランジスタは拡散接点が少ないためスペースが小さくてすむ）が多いため、チップ面積が小さいということである。入力バス、BitInH 3604からのデータビットは、WriteToMasterH 3606とWriteToMasterL 3608からなる差分信号ペ

アを表明することでフリップフロップ3602のマスターハーフに書き込まれる。クロックサイクルのもう半分の間、ShiftToSlaveH3610とShiftToSlaveL3612からなる差分信号ペアを表明することで、フリップフロップ3602のスレーブハーフにデータビットを保存する。前のワードの対応するビットからのデータビットは、ShiftToMasterH3616とShiftToMasterL3618からなる差分信号ペアを表明することで、ShiftDataInH3514を介してフリップフロップ362のマスターハーフにシフトされる。保存されたデータビットは、ReadWord3620を表明してセル3600から読み取られ、これが逆ビット値をBitOutL3622バスに入れる。セル3600のバランスが比較を実行し、前述のセル3500に似た方法でチェーンを使用可能にする。XORサブ回路は保存されたビット値を（フリップフロップ3602のマスタ値とスレーブ値が互いの補数であるためこれらを利用して）BitInH3604とBitInL3624の相補値と比較する。

図37は、フィールドセットを形成するCMOS SMCCAMビットセル配列を示す。信号、ShiftDataInH3514とShiftDataInL3516は、第1ワード900がデータを900にシフトできないため、第1ワード900にグラウンドされる。

符号付き数をSMCCAMに保存しなければならない場合、読み取りおよび書き込みプロセスの両方で、符号ビット（すなわち「1

」が正を暗示）を逆転させるだけである。このようにすると、ハードウェアで正の数が負の数より「大きく」なる。別のアプローチとして、符号付き数を保存・照会しなければならない場合各フィールドの符号ビット位置の論理を変更するものがある。浮動小数点数は、ビット位置のより上位のセットに指数を入れ、指数を正の数として保存することで容易に保存される。故に、IEEE浮動小数点標準数は正しく処理される。

ラスタライン内で縦の複数スパン

前述のセグメント・スパン・ラスタリゼーション方法2100では、各多角形はあるラスタラインに1つしかスパンを持つことができず、そのスパンはスクリ

ーン座標の $x-y$ 平面に平行なラインセグメント（故に、上記名 2100）で表される。スパンのエンドポイントの位置はオブジェクト精密であるため、ピクセルの境界内のどこにでも水平に位置づけることができる。スパン端部がピクセル幅の 20% をカバーしているとする、そのスパンはそのピクセルのカラー値の 20% となり、ピクセル幅の残りの 80% のカラー値とブレンドされる。これによって水平のアンチエイリアシングができるが、縦のアンチエイリアシングにはならない。すなわち、ブレンドは縦ではなく水平に起こる。多角形の辺が縦より水平に近いと、その辺は複数のピクセル領域を横切る可能性が高いため、その多角形のカラーを複数ピクセルにブレンドしなければならない。しかしながら、1つのラインセグメントを使ってスパンを表現すると、多角形のカラーは 1 ピクセル（セグメントスパンエンドポイントを含むピクセル）にブレンドされ、両側のピクセルは 100% カバーされていると想定されるためブレンドされない（可視スパンポーションが 1 ピクセルより広いと想定）。縦のアンチエイリアシングの欠如は一部アプリケーションでは許容可能だが、高品質画像は両方

向でアンチエイリアシングしなければならない。

縦のアンチエイリアシングを追加する方法として、各ラスタラインを複数の「サブラスタ」ラインに縦に分割するものがある。そして、セグメント・スパン・ラスタリゼーション方法 2100 を各サブラスタに用いて、縦と水平にブレンドすることでピクセルカラー値を生成することができる。図 38A にラスタライン 3802 とそのピクセル 3804、多角形の 2 辺 3806 と 3808、その多角形の 2 つのサブラスタスパン 3810 と 3812 を示す。図 38B に、その多角形の 4 つのサブラスタスパン、3814、3816、3828 および 3820 を示す。可能なバリエーションとして、ピクセルが同じ多角形のこのようなサブラスタスパン 2 つのみにカバーされている場合、ピクセル中央で 1 つのカラー値を計算することが可能となる。

サブラスタラインを用いることで縦のエイリアシングは減るが、無くなることはない。多くのアプリケーションにとってこれは許容可能な近似である。

台形スパン・ラスタリゼーション方法

スパンは、1次元のラインセグメントではなく2次元の台形で表すことができる。ラインセグメントもプレーナ台形も3次元のオブジェクト精密の視点座標である。台形表現の利点は、辺情報をラスタライン内に保存することによって水平と縦の両寸法で正しいアンチエイリアシングを行う点である。

図39に、2つの三角形3902と3904、1つのラスタライン3906をディスプレイスクリーン104内に持つ三角形メッシュを示す。各三角形3902と3904のラスタライン3906内の領域は台形形状スパンで、 S_{11} 3908と S_{12} 3910というラベルが付いている。台形スパンにおいて、スパンのエンドポイ

ントは台形の辺であり、3次元スクリーン座標において次の2つのポイント：ラスタライン上辺の1つと、ラスタライン下辺の1つで記述される。故に、台形スパンはスクリーン座標の4つのポイントで記述される。例えば、図の左台形スパン3908のx-z空間のエンドポイントは次によって記述される：1) (x_{11LT}, z_{11LT}) に位置する左エンドポイント上部、2) (x_{11LB}, z_{11LB}) に位置する左エンドポイント下部、3) (x_{11RT}, z_{11RT}) に位置する右エンドポイント上部、および4) (x_{11RB}, z_{11RB}) に位置する右エンドポイント下部。これら4つのポイントは実際にx-z平面にあるが、y座標はラスタライン3906の上または下位置で決まる。座標変数のサブスクリプトが次の3つのことを示すのに注意する：1) スパン番号（すなわち、「11」）、2) 左または右エンドポイント（すなわち、「L」または「R」）、および3) ラスタライン上または下（すなわち、「T」または「B」）。

不透明台形スパンは、その長さの大半でより遠い幾何形状を閉塞するが、別の台形スパンと隣接しないエンドポイント3912と3914では、より遠い幾何形状は部分的に可視である。しかしながら、図39に示すような隣接する台形スパンでは、共通エンドポイント3916はより遠い幾何形状を閉塞する。台形スパンを用いる時これが問題となる。すなわち、隣接しない台形スパンのエンドポイントはより遠い幾何形状を閉塞しないが、隣接するエンドポイントは共により遠い幾何形状を閉塞する3918。本発明は、隣接するスパンを検索することで

この問題を解決することに成功している。

図40に、同じラスタラインの台形スパングループを示すが、これらスパンは図11のセグメントスパンの台形バージョンである。図40において、スパンの可視ポーションはシェーディングされて

いる。多角形メッシュのスパン（すなわち隣接するスパン）で、スパン上部が互いにエンド・ツー・エンドで重なり、スパン下部も同様であることに注意する。スパンエンドポイントは、ビューイングポイントから遠いスパンに辺の「シャドウ」を投影する。例えば、 S_2 の左エンドポイント4002は、 S_1 にこのようなシャドウ4004を投げかけている。

台形スパンを用いる際のもう1つの懸念は、多角形がラスタライン上下の両方でなく、上か下と交差する時に発生する。これは一般に次のところで発生する：1) 多角形の角、2) 多角形の水平辺、または3) 閉塞スパンがスパンの裂辺のみ可視に残す。この1例を図40に示すが、ここで S_0 の右端エンド4006は可視である。このような状況で、台形スパンは実際に三角形のような形状であるため、上または下の長さがゼロの「縮重台形」とみなされることがある。

本書ではセグメント・スパン・ラスタリゼーション方法2100を詳しく説明してきたが、台形スパン・ラスタリゼーション方法は、最前スパンを左から右に追い、スパンの可視ポーションをラスタリゼーション・プロセッサ512に送るという点で類似である。重要な違いは、台形スパンの隣接しないエンドポイントがより遠い幾何形状を完全に閉塞しない点である。台形スパン・ラスタリゼーション方法はこの種のスパンを正しく取り扱わなければならない。

台形形状スパンの取り扱いにはいくつか方法がある。本書で提示する台形スパン・ラスタリゼーション方法の4つの選択肢のそれぞれについて方法フロー図を示すのではなく、対応するSOT照会の選択肢と共にそれぞれの特徴を説明する。これら4つの選択肢は、本書で提示するラスタリゼーション方法をいかに台形形状スパンの処理に拡大するか図示するよう選んだ。

バージョン1：スパン上部とスパン下部に別個の閉塞処理

台形スパンは、次の集合として記述できる： 1) ラスタライン上部に沿った台形スパンの境界上のラインセグメントであるスパン上部、 2) ラスタライン下部に沿ったラインセグメントであるスパン下部、および 3) スパン上部とスパン下部の間の領域。本書セクションに説明する台形スパン・ラスタリゼーション方法のバージョンは、スパン上部についてスパン下部とは別に閉塞処理を行う。同じラスタラインでは、スパン上部とスパン下部は互いを閉塞することがないのでこれが可能である。スパン上部の閉塞処理は、セグメント・スパン・ラスタリゼーション方法 2 1 0 0 でセグメントスパンで行った閉塞処理と同じ方法で行うことができる。スパン下部の閉塞処理は、同じ方法で別に行うことができ、上部と下部のペアをピクセルカラー生成のため正しくペア組みするため、スパン上部の閉塞処理と平行して行わなければならない。

台形スパン・ラスタリゼーション方法のこのバージョンは、次の「スパン状のもの（該当するエンドポイント含む）を記録する： 1) どの部分をレンダリングするか確認するため現在テスト中の台形スパンである、現在のスパン、 S_P 、 2) 可視性をテスト中の S_P の上辺、カレントスパン・トップ、 S_{CT} 、 3) 可視性をテスト中の S_P の下辺、カレントスパン・ボトム、 S_{CB} 、 4) スパンメモリ 4 0 8 から最近読み取った台形スパンである、新しいスパン、 S_N 、 5) S_N の上辺、新しいスパン・トップ、 S_{NT} 、 6) S_N の下辺、新しいスパン・ボトム、 S_{NB} 、 7) 前述のルール 4 に類似の状況で S_{CT} が S_{NT} と交差するため、一時的にセーブした台形スパン、トップ・セーブ・スパン、 S_{ST} 、および 8) S_{CB} が S_{NB} と交差するため、一時的にセーブした台形スパン、ボトム・セーブ・スパン、 S_{SB} 。スパン上部はスパン下部と別に処理し、これらが別のスパンと

の交点を持つ可能性があるため、2つの異なるセーブしたスパンが必要となる。ラスタリゼーション中により多くの情報を記録するには、照会プロセッサ 5 1 0 内のスパン・パラメータ・レジスタ 3 1 0 4 にレジスタを追加する必要がある。

図 4 1 に、図 4 0 のスパン例、 S_1 4 1 0 1 と S_2 4 1 0 2 を拡大したバージョンを示し、 S_1 4 1 0 1 を現在のスパン、 S_P と想定する。 S_{1T} 4 1 0 3 と呼ぶ S_1 4 1 0 1 のスパン上部はその対応する SOT 領域 4 1 0 4 を持ち、それ 4

103を閉塞する S_{2T} 4106を見つける。このSOT領域4104はスパン上部のみ検索しているため、 S_{2B} 4108は見つけない。同様に、 S_1 のスパン下部、 S_{1B} 4110のSOT領域4112は、それ4110を閉塞する S_2 4108を見つける。

図42に、 S_{31} 4201から S_{39} 4209のラベルの付いた、ラスタライン4200内の部分的スパンセットの図を2つ示す。上の図は、スパンの $x-z$ 図（ルック方向は y 軸と平行）、下の図は、同じスパンの $x-y$ 図（ルック方向は z 軸と平行）である。下の図で、スパンの隠れたポーションは図示しないため、そのラスタライン4200についてスパンの無限精密レンダリングがどのように描かれるかを表す。しかし、ディスプレイ装置には無限の精密がないため、スパンカラーは各ピクセルについて正しい割合でブレンドしなければならない。図では、スパン上部と下部の可視ポーションは太い線で示す。スパン上部と下部の可視ポーションはラスタリゼーション・プロセッサ512に送られ、ここで多角形辺導関数と共にピクセルカラーを決定する。図42のスパンは、スパン上部またはスパン下部のみ可視のスパンや、ラスタライン4200の高さ全体をカバーしないスパンを含む各種ケースの取り扱いを示すように選んだ。

セグメント・スパン・ラスタリゼーション方法2100の場合同様、スパンは左から右へ処理する。図42の例は、ラスタライン中間（すなわち、水平）で始まり、 S_{31} 4201が S_P 、 S_{31T} 4211が S_{CT} 、 S_{31B} 4221が S_{CB} である。 S_{CT} と S_{CB} のSOT照会では何も見つからず、 S_N が無効にセットされ、セグメント・スパン・ラスタリゼーション方法2100のルール2 2300に類似のレンダリングルールによって、 S_{31} 4201をその右エンドポイントまでレンダリングする。 S_{31} 4201のレンダリングは、 X_{31RT} 4251を含むピクセルまでピクセルの100%をカバーする。このポイントから、対応する多角形辺の辺導関数（式1参照）を使って、各ピクセルのパーセントカバレッジを決定し、このパーセンテージでピクセルカラーを重みづけする。スパン4201は、エンドポイントの全長（台形スパンのエンドポイントはラインセグメント）が可視と保証されているため、右エンドポイントまでレンダリングできる。再びル

ール2 2300類似により、隣接するスパン上部およびスパン下部の照会で、 $S_{32T} 4212$ と $S_{32B} 4222$ が見つかる。方法のこのポイントで、 $S_{32} 4202$ は S_P 、 $S_{32T} 4212$ は S_{CT} 、 $S_{32B} 4222$ は S_{CB} である。 S_{CT} と S_{CB} のSOT照会では何も見つからず、隣接するスパンも見つからない。そのため、 S_N を無効にセットする。

ルール2 2300類似のルールによって、 $S_{32} 4202$ の全体をレンダリングし（その右エンドポイントは可視保証）、再び辺導関数を使ってピクセルカバレッジを計算する。例を見ると、 $X_{31RT} 4251$ と $X_{31RB} 4252$ の間に少なくとも部分的に含まれるピクセルについて、 $S_{31} 4201$ と $S_{32} 4202$ によるパーセントカバレッジの合計は100%である。隣接するスパン上部と下部を探す照会は失敗で、完全に $X_{32RB} 4252$ の左にあるスパン

はすべて無効とし、 S_P と S_N の両方を無効とセットする。

ルール1 2200に類似のルールによって検索を行い、 $X_{32RT} 4253$ と $X_{32RB} 4254$ の最前スパン上部と下部を探し、 $S_{33} 4203$ が見つかり、これを S_P に割り当て、 S_{CT} を $X_{32RT} 4254$ の左の $S_{33T} 4213$ の部分にセットし、 S_{CB} を $X_{32RB} 4253$ の左の $S_{33B} 4223$ の部分にセットする。 S_{CT} と S_{CB} のSOT照会で $S_{34T} 4214$ と $S_{35B} 4225$ （これらは異なるスパンからのものであることに注意）が見つかり、それぞれ S_{NT} と S_{NB} に割り当てる。 S_N を $S_{35} 4205$ ではなく $S_{34} 4204$ にセットするが、 $S_{34} 4204$ の可視ポジションがより左に遠いため、方法は左から右に進む。

ルール3 2400に類似のルールで $S_{33} 4203$ を部分的にのみレンダリングする。この時点で、 4203 の $X_{34LT} 4225$ の右の領域はまだその可視性を決定しなければならないため（一般に S_{NT} と S_{NB} が同じスパンからのものではない場合）、ラスタライゼーションプロセッサ512は 4203 を $X_{34LT} 4225$ までしかレンダリングできない。 $X_{34RT} 4255$ はほぼ確実にピクセル辺と揃っていないため、 $X_{34RT} 4255$ を含むピクセルは、それ（と後で追加した残り）をカバーするとわかっている $S_{33} 4203$ のポジションで部分的にカラー化される。または、 $S_{33} 4203$ によるそのピクセルへのカラーコントリビ

ューション全体を、ピクセルカバレッジの全体量がわかるまで延期することができる。この方法のこの時点で、 $S_{34} 4204$ は S_P 、 $S_{34T} 4214$ は S_{CT} 、 $S_{35B} 4225$ は S_{CB} である。 S_{CT} （すなわち、 $S_{34T} 4214$ ）のSOT照会で $S_{35T} 4215$ が見つかり、これを S_{NT} にする。この S_{NT} が前のステップの S_{NB} と一致するので（すなわち、これらが同じスパン、 $S_{35} 4205$ から）、 S_{CB} （すなわち、 $S_{35B} 42$

25）のSOT照会を方法の次のステップまで延期する。 S_N を $S_{35} 4205$ にセットする。

この方法中、 S_{NT} と S_{NB} が同じスパンからのものでない時、ラスタラインの上部に沿った現在の x 値位置からラスタラインの下部に沿った現在の x 値位置の最左までしかレンダリングできない。2つの x 値の最左で、左のスパンはすべて考慮されているため、レンダリングができる。

ルール3 2400に類似のルールで、 $S_{34} 4204$ をラスタリゼーション・プロセッサ512でレンダリングする。この時、 S_{NT} と S_{NB} は同じスパンからのものであるため、 $S_{34} 4204$ を $S_{35} 4205$ の左エンドポイントまでレンダリングする。また、 $S_{33} 4203$ は、その可視性が決定されたので、そのレンダリングを完了できる。この方法のこの時点で、 $S_{35} 4202$ を S_P に、 $S_{35T} 4215$ を S_{CT} に、 $S_{35B} 4225$ を S_{CB} にする。 S_{CT} （すなわち、 $S_{35T} 4215$ ）のSOT照会で $S_{37T} 4217$ が見つかり、これを S_{NT} にする一方、 S_{CB} （すなわち、 $S_{35B} 4225$ ）のSOT照会では何も見つからず、 S_{NB} を無効にし、 S_N を $S_{37} 4207$ にセットする。

ルール3 2400に類似のルールにより、 $X_{35RB} 4256$ が $X_{35RB} 4256$ と $X_{37LT} 4257$ の間の最左選択であるため、 $S_{35} 4205$ をその左エンドポイントから $X_{35RB} 4256$ までレンダリングする。 S_{CB} （すなわち、 $S_{35B} 4225$ ）の隣接スパン下部の検索で $S_{36B} 4226$ が見つかる。この方法のこの時点で、 $S_{35} 4206$ は S_P 、 $S_{37T} 4217$ は S_{CT} 、 $S_{36B} 4226$ は S_{CB} 、 $S_{37T} 4217$ は S_{NT} であり、 S_{CB} （すなわち、 $S_{36B} 4226$ ）のSOT照会で $S_{39B} 4229$ が見つかり、これを S_{NB} にする。

ルール3 2400に類似のルールにより、 $x_{37LT} 4257$ が $x_{37LT} 4257$ と $x_{39LB} 4258$ の間の最左選択であるため、 $S_{36} 4206$ を $x_{37LT} 4257$ までレンダリングする。また、 $S_{35} 4205$ のレンダリングは $x_{37LT} 4257$ まで継続できる。 $S_{37} 4207$ は隠れておらず、1つのスパン上部しか持たない（スパン下部なし）ことがわかっているため、 4207 を S_P にする。この時点でまた、 $S_{37T} 4217$ はまだ S_{CT} 、 $S_{39B} 4229$ は S_{CB} であり、 $S_{37} 4207$ に隣接するスパンを見つける照会で何も見つからないため、 S_{NT} は無効で、 $S_{39B} 4229$ がまだ S_{NB} である。

ルール2 2300に類似のルールにより、 $S_{37} 4207$ の全体をレンダリングする。 $S_{35} 4205$ も $S_{36} 4206$ も、 $x_{37RT} 4259$ （ $S_{35} 4205$ を完了）が $x_{37RT} 4259$ と $x_{39LB} 4258$ の間の最左選択であるため、ここまでレンダリングすることができる。この時点で、 $S_{38} 4208$ は S_P 、 S_{CT} は無効、 $S_{39B} 4229$ は S_{CB} 、 S_{NT} は無効で、 S_{CB} （すなわち、 $S_{39B} 4229$ ）のSOT照会で何も見つからないため、 S_{NB} は無効である。

ルール1 2200に類似のルールにより、検索を行って、 $x_{37RT} 4259$ の最前スパン上部を見つける。この検索で $S_{38T} 4218$ が見つかり、 S_{CT} を $x_{37RT} 4259$ の右の $S_{38T} 4218$ のポジションにセットする。 S_{CT} のSOT照会で $S_{39T} 4219$ が見つかり、これを S_{NT} に割り当てる。この時点で、 $S_{38} 4208$ はまだ S_P 、 $S_{39B} 4229$ はまだ S_{CB} で、 S_{NB} はまだ無効である。

ルール3 2400に類似のルールにより、 $S_{38} 4208$ を、 $S_{39B} 4229$ の左エンドポイントまでレンダリングする。 $S_{39} 4209$ は次の S_P になるため、他方の残りの完了していないスパ

ン、 $S_{36} 4206$ を、 $S_{39B} 4229$ の左エンドポイントまでレンダリングして完了する。すべての多角形辺導関数を正しく考慮するよう注意する。

ラスタライン4200処理は $S_{39} 4209$ を S_P として続くが、我々の例の右辺に達した。この例の左および右境界内に完全に隠れたスパンがあったなら、これらはSOT照会で全く見つからないか、ルール5 2600類似のレンダリングルールによって放棄されることによって完全に回避されている。同様に、2つ

のスパン上部または2つのスパン下部の間に交差があったなら、ルール4 25 00に類似のルールが適用され、セーブしたスパンが用いられている。

スパン上部もスパン下部も持たない（高さが1ピクセルより小さい）か、スパン上部もスパン下部も隠れているが、ラスタライン内で可視（2つのほぼ水平な多角形辺の間からのぞく）のスパンの取り扱いについて特殊なケースを追加することができる。大半の実用アプリケーションでは、このような限界的に可視のアイテムは無視できる。

いくつかの異なるスパンからのカラー情報を特定のピクセルについてブレンドすることが可能である。このような状況では（および2つのスパンのカラーのみのブレンドでも）、高品質画像を作るには妥当なパーセントカバレッジ近似のみ必要である。

ハードウェアにインプリメントする場合、台形スパン・ラスタリゼーション方法のこのバージョンでは、各スパン上部に1つのSMCCAMワード900、各スパン下部に別に1つを用いる。スパン上部もスパン下部もスパンメモリ408の同じページに保存できる。これは、スパンメモリ408を2つの本質的に独立した閉塞プロセスの間で時分割することを意味し、スパン上部とスパン下部を別

のページに保存する場合に比べてパフォーマンスが落ちる。閉塞に1つのページを使うことで、図39に示すように、スパンメモリ408はスパンをダブルバッファするのに合計2ページを必要になる。

スパンメモリ408で2ページ用いると、2つの閉塞プロセスを平行して行うことができる。しかしながら、スパン導関数フィールド812や情報ポインタフィールド812など一部のフィールドは両方の閉塞プロセスが必要とするため、重複させる必要がある。また、閉塞処理に2ページを用いることは、スパンメモリ408に合計4ページ必要なことを意味することがある。しかし、非インターレースディスプレイ装置を用いる場合は、カレントラスタラインのスパン下部セットは次のラスタラインのスパン上部セットと同じである。そのため、1ページを書き込み、2ページを読み取りに使い、これら2ページの内1頁を次のラスタラインに再使用するため、スパンメモリ408には合計3ページ必要となる。

インターレースディスプレイ装置を用いる場合、同じ技術を使えるが、ラスタラインのレンダリング順はディスプレイ装置と一致しない。

もう1つの代替え案は、他のメモリに可視下部スパンポーション（すなわち閉塞処理後）のリストをセーブして、これらを次のラスタラインのスパン上部に用いるものである。このようなメモリは、ラスタリゼーション・プロセッサ512に含めることができる。これは確実に可能で、可視上部スパンの再計算ではおそらくより経済的である。

バージョン2：ラスタラインの各トランジションでSOT照会を実行

台形スパン・ラスタリゼーション方法のこのバージョンでは、スパンは左から右に処理するが、閉塞処理では各スパンを別個のスパ

ン上部とスパン下部ではなく、1つのユニットとして扱う。このため、SOT領域をスパン上部またはスパン下部で可視性トランジションによって右と左に境界をつける。

図43に、図40の例のスパン S_1 4301と S_2 4302の拡大版を示す。この例のスタート時、 S_1 4301は現在のスパン、 S_P で、 x_{ILB} 4304が現在の x 座標、 x_{CL} である。そのため、ラスタラインは S_1 4301の x_{ILT} 4303と x_{ILB} 4304の間の部分を含む x_{ILB} 4304までレンダリングされている。カレントスパン、 S_c は同じ可視性特徴を持つ x_{CL} （すなわち、 x_{ILB} 4304）の右の S_P （すなわち、 S_1 4301）の部分、すなわち、 x_{ILB} 4304と x_{IRT} 4306（ x_{IRT} 4306を x_{CR} の値とする）の間の部分である。 S_1 4301の可視性特徴は、スパン4301が右でより遠い幾何形状を閉塞しないが、そのポイント4306の左で閉塞するため、 x_{IRT} 4306で変化する。この S_c のSOT領域4308を図43に示し、式123によって記述する。

(式 123)

$$\begin{aligned}
 F_{nQ} = & F_{nV} \wedge F_{nQ} \wedge (n \neq S_c) \wedge [(x_{ALT} < x_{CR}) \vee (x_{ALB} < x_{CR})] \wedge [(x_{ART} < x_{CL}) \vee (x_{ARB} < x_{CL})] \\
 & \wedge [[(z_{ALT} < z_{CFT}) \vee (z_{ALB} < z_{CFB})] \vee [(z_{ART} < z_{CFT}) \vee (z_{ARB} < z_{CFB})]] \\
 F_{nQ} = & F_{nV} \wedge F_{nQ} \wedge (n \neq 1) \wedge [(x_{ALT} < x_{IRT}) \vee (x_{ALB} < x_{IRT})] \wedge [(x_{ART} < x_{ILB}) \vee (x_{ARB} < x_{ILB})] \\
 & \wedge [[(z_{ALT} < z_{temp1}) \vee (z_{ALB} < z_{ILB})] \vee [(z_{ART} < z_{temp1}) \vee (z_{ARB} < z_{ILB})]]
 \end{aligned}$$

この照会では、2つの異なる z 値、 z_{CFT} と z_{TFB} を用いるが、閉塞テストするスパンポーションがその上部と下部に異なる深さ値を持つためである。 z_{CFT} の値は S_c の x 座標間隔内のスパン上部の最大 z 値にセットし、これは z_{temp1} 4309 のラベルがついた、値 x_{ILB} 4304 を持つ x 座標である。値 z_{temp1} 4309 は、 S_1 4301 のスパン導関数を使って計算する。同様に、 z_{CFB} の値は、 S_c の x 座標間隔内のスパン下部の最大 z 値にセットし、こ

の値は z_{ILB} 4310 である。SOT 照会で S_2 4302 が見つかり、4302 を S_N にする。

セグメント・スパン・ラスタライゼーション方法 2100 のルール 3 2400 に類似のルールによって、 S_P (すなわち、 S_1 4301) を x_{CL} (すなわち、 x_{ILB} 4304) から S_N (すなわち、 S_2 4302) の最左ポイントまでレンダリングするが、これは x_{2LB} 4304 である。この間隔中、可視性特徴はラスタラインについて変化しない。すなわち、 S_1 4301 はこの間隔内で可視の唯一の幾何形状である。

x_{CL} の値は x_{2LB} 4311 に変更され、 S_c は、同じ間隔で S_2 4302 の下辺と組み合わされ、 x_{2LB} 4311 から x_{2LT} 4312 まで S_1 4301 の上辺である。 S_N の有効は無効にセットする。この S_c について、全幅で S_2 4302 に徐々にカバーされる S_1 4301 を含む点から可視性特徴は一定である。対応する照会領域 4313 は x 座標寸法で S_c の間隔にある。しかし、 z 座標寸法では、スパン上部座標をテストして、これらが z_{temp2} 4314 の前にあるか確かめ、スパン下部座標をテストして、これらが z_{2LT} 4316 の前にあるか確かめる。座標 z_{temp2} 4314 は x 座標 x_{ILB} 4304 で S_1 4301 のスパン上部の z 座標である。左上角がこの範囲にあるスパンは可視になるため、この照会領域 4313 には z_{temp2} 4314 と z_{2LT} 4316 の間のスパン上部の検索を含める必要がある。照会オペレーションは式 124 に示す通りである。

(式 124)

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq S_C) \wedge [(x_{nLT} < x_{CR}) \vee (x_{nLB} < x_{CR})] \wedge [(x_{nRT} < x_{CL}) \vee (x_{nRB} < x_{CL})] \\
 &\quad \wedge [(z_{nLT} < z_{CPT}) \vee (z_{nLB} < z_{CFB})] \vee [(z_{nRT} < z_{CPT}) \vee (z_{nRB} < z_{CFB})] \\
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq 1) \wedge (n \neq 2) \\
 &\quad \wedge [(x_{nLT} < x_{2LT}) \vee (x_{nLB} < x_{2LT})] \wedge [(x_{nRT} < x_{2LB}) \vee (x_{nRB} < x_{2LB})] \\
 &\quad \wedge [(z_{nLT} < z_{temp2}) \vee (z_{nLB} < z_{2LB})] \vee [(z_{nRT} < z_{temp2}) \vee (z_{nRB} < z_{2LB})]
 \end{aligned}$$

この例で、照会領域 4 3 1 3 では何も見つからないため、 S_N は無効のままである。

ルール 2 2 3 0 0 に類似のルールによって、 S_c は隠れていないため、レンダリングし、 S_c の全幅について S_1 4 3 0 1 と S_2 4 3 0 2 がブレンドされる。そしてこのルールで隣接するスパンの等価を「チェック」し、 S_2 4 3 0 2 が x_{2LT} 4 3 1 2 から右に続くため、存在することがわかる。そのため、 S_c を S_2 4 3 0 2 の x_{2LT} 4 3 1 2 と x_{2RT} 4 3 1 8 の間の部分にセットし、 x_{CL} を x_{2LT} 4 3 1 2 にセットする。再び（この方法のこのバージョンでは常にそうだが）、 S_c はその全幅にわたって一貫した可視性特徴を持つ。この S_c の SOT 領域 4 3 2 0 を図 4 3 に示し、式 1 2 5 で記述する。値、 z_{temp3} 4 3 2 2 は、 S_2 4 3 0 2 の x 座標 x_{2LT} 4 3 1 2 でのスパン下部の z 値である。

(式 125)

$$\begin{aligned}
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq S_C) \wedge [(x_{nLT} < x_{CR}) \vee (x_{nLB} < x_{CR})] \wedge [(x_{nRT} < x_{CL}) \vee (x_{nRB} < x_{CL})] \\
 &\quad \wedge [(z_{nLT} < z_{CPT}) \vee (z_{nLB} < z_{CFB})] \vee [(z_{nRT} < z_{CPT}) \vee (z_{nRB} < z_{CFB})] \\
 F_{nQ} &= F_{nV} \wedge F_{nQ} \wedge (n \neq 2) \wedge [(x_{nLT} < x_{2RT}) \vee (x_{nLB} < x_{2RT})] \wedge [(x_{nRT} < x_{2LT}) \vee (x_{nRB} < x_{2LT})] \\
 &\quad \wedge [(z_{nLT} < z_{2LT}) \vee (z_{nLB} < z_{temp3})] \vee [(z_{nRT} < z_{2LT}) \vee (z_{nRB} < z_{temp3})]
 \end{aligned}$$

このラスタラインのスパンのラスタリゼーションはこのポイントから続くことになる。この方法のこのバージョンにも、ルール 1 2 2 0 0、ルール 4 2 5 0 0 およびルール 5 2 6 0 0 に類似のルールがある。

一般に、この方法のこのバージョンでは、スパンを取り、これら

を水平に細分して一貫した可視性特徴を持つセクションを作る。これは、スパンのエンドポイント領域（図 3 9 のエンドポイント 3 9 1 2 と 3 9 1 4 のように、遠い幾何形状を閉塞しない部分）をスパンの中央ポーシオンと別に扱うということの意味する。これによってより多くの照会オペレーションが必要となるが、1

つの閉塞プロセスで行うことができる。

台形スパン・ラスタリゼーション方法のこのバージョンでは、1つのエンドポイントにつき2つではなく4つのフィールドがあるため、1つのSMCCAMワード900により多くのフィールドが必要となる。ワード番号フィールド802が、2つの台形スパンの組み合わせであるカレントスパンとの不等価をチェックできなければならないため、ワード番号フィールド802に比較論理も追加する必要がある。上記の例では、 X_{2LB} 4310から X_{2LT} 4312までをカバーする照会領域4313に対応するSCの値は、 S_1 4301と S_2 4302の組み合わせである。そのため、SOT照会では、 $(n \neq 1)$ と $(n \neq 2)$ の両方をチェックしなければならない、その照会オペレーション中に2つの不等価比較が必要となる。この方法のこのバージョンには、スパンを保存するのにワード900を1つしか必要としないという利点がある。

バージョン3：より複雑なSOT領域の可能な1ワードあたりより多くの比較

台形スパン・ラスタリゼーション方法のこのバージョンでは、閉塞処理はバージョン2同様、各スパンをユニットとして扱う。しかしながら、このバージョンでは各SMCCAMワード900でより多くの比較オペレーションを実行することによって、より複雑なSOT領域を作ることができる。各可視性トランジションでスパンを分割するのではなく、カレントスパン、 S_c には1つの可視性トラ

ンジションを入れることができる。本質的に、このバージョンのSOT領域はバージョン2で定義する2つのSOT領域の合併である。図44に、図43に示すものと同じ2つのスパン4401と4402、およびこの方法のバージョン2からの2つのSOT領域4313と4320の合併であるSOT領域4404を示す。このSOT照会を実行するため、照会オペレーションは式124と式125の右辺の論理和（すなわち、論理「or」）である。

より複雑な照会ファンクションを実行してSOT領域4404を生成するには、各SMCCAMワード900のフィールドがそれぞれ2つの比較オペレーションを平行して行う必要がある（2つのオペレーションを順次行くと、これはバージョン2と同じになる）。これによってハードウェアはより複雑になるが、閉塞

処理を実行する時間が約2倍短くなる。

バージョン4：閉塞領域のよりルーズな近似の使用

閉塞処理時間を短縮するもう1つの選択は、閉塞領域のよりルーズな近似を使用することである。例えば、図44の「L字形」SOT領域4404（十分な近似）は、その領域を増やすことで単純な矩形領域とすることができる。図45に、図44に示すのと同じ2つのスパン4501と4502、この方法のバージョン3からのSOT領域4404より大きくて単純なSOT領域4504を示す。このSOT領域4504を生成するSOT照会は、式124と式15を各フィールドにさらにルーズな制約を使うように組み合わせたものである。照会オペレーションは次の通りである：

$$\begin{aligned}
 F_{AQ} = & F_{AV} \wedge F_{AQ} \wedge (n \neq SC) \wedge [(x_{ALT} < x_{CR}) \vee (x_{ALB} < x_{CR})] \wedge [(x_{ART} < x_{CL}) \vee (x_{ARB} < x_{CL})] \\
 & \wedge [(z_{ALT} < z_{CFT}) \vee (z_{ALB} < z_{CFB})] \vee [(z_{ART} < z_{CFT}) \vee (z_{ARB} < z_{CFB})] \quad (\text{式 126}) \\
 F_{AQ} = & F_{AV} \wedge F_{AQ} \wedge (n \neq 1) \wedge (n \neq 2) \wedge \\
 & \wedge [(x_{ALT} < x_{2RT}) \vee (x_{ALB} < x_{2RT})] \wedge [(x_{ART} < x_{2LB}) \vee (x_{ARB} < x_{2LB})] \\
 & \wedge [(z_{ALT} < z_{2mp2}) \vee (z_{ALB} < z_{2LB})] \vee [(z_{ART} < z_{2mp2}) \vee (z_{ARB} < z_{2LB})]
 \end{aligned}$$

このルーズな近似をSOT領域4504に用いると、実際には隠れている潜在的閉塞スパンが（平均して）より多く見つかるので、パフォーマンスが落ちる。しかしながら、照会オペレーション数が減るので、正味ゲインとなる場合がある。

透明性

スパン・ソーティング・レンダリング・アーキテクチャ500は、不透明な幾何形状と同じ方法でスパンを生成することにより、透明な幾何形状（見透かせる着色幾何形状）をレンダリングすることができる。半透明は同様に扱えるが、本書では透明性に注目する。ラスタリゼーションプロセスを支援するため、各スパンの透明性を次の3つの方法の1つでスパンメモリ408にマークする：1）別のフラッグビット、 F_{AT} を各ワード900に追加する、2）情報ポインタフィールド814のビット、または3）別のフィールド、半透明フィールドを各ワード900に追加する。不透明係数（すなわち、透明幾何形状が透過する光の量）は、他のカラー情報と共に多角形情報メモリ514に保存される。

本書に提示するラスタリゼーション方法で透明スパンを取り扱う方法には各種

ある。以下の説明をわかりやすくするため、図46（図11の修正版）にラスターのセグメントスパン4600から4613のセットを示すが、ここで可視不透明面は太い黒線で、透明スパン4602、4605および4607の可視ポーションは太い隠線で示す。本書のこのセクションではセグメントスパンの例を用いるが、透明方法は台形スパンにも四辺形スパンにも適用できる。

バージョン1：可視性特徴をカレントスパンに対して一定に保つ

この方法のこのバージョンでは、カレントスパン、 S_c を透明スパンが閉塞する場合、必要なら S_c を小さくして、 S_c の可視性特

徴を一定に保つ。この方法は最前の不透明スパンを追い、これらが S_c を閉塞していたら、透明スパンの効果を加える。

例えば、図46で、 S_1 4601のSOT照会で透明スパン S_2 4602が見つかったら、 S_c は x_{1L} 4624と x_{2L} 4626の間の S_1 4601の部分に変化し、 S_1 4601もセーブしたスパン、 S_s として保持され、その x_{2L} 4624の右の部分が次の S_c になる。この部分が S_c （すなわち、 x_{2L} 4626と x_{3L} 4628の間の S_1 4601の部分）になったら、そのSOT照会も S_2 4602を見つけ、 S_1 4601と S_2 4602の組み合わせがレンダリングされる。 S_3 4603と S_2 4602の組み合わせが同様に処理される。

S_4 4604に達したら、その x_{4L} 4630と x_{5L} 4632の間の部分が先に処理され、 S_2 4602の効果を含める。そして、 S_c が x_{5L} 4632の右の S_4 4604の部分になると、SOT照会で S_5 4605と S_6 4606の両方が見つかる。この時点で、この方法および装置を次のいずれかを行うように設計することができる：1) 最左不透明閉塞スパンを読み取る、2) 最左透明閉塞スパンを読み取る、または3) 両方のタイプの最左スパンを読み取る。ここでは、これら選択肢の内1番目を想定する。故に、ルール3 2400に類似のレンダリングルールによって、 S_c が x_{5L} 4632と x_{6L} 4634の間の S_4 4604のポーションまで減少する。しかし、 S_c をレンダリングする前に、SOT領域の透明スパンを読み取って、 S_5 4605を取得する。 S_5 4605は S_c （すなわち、 x_{5L} 4632と x_{6L} 4634の間の S_4 4604）のすべてをカバ

一しているため、SCはこれ以上分割されず、 x_{5L} 4632から x_{6L} 4634までの間隔で S_4 4604と S_5 4605の組み合わせとしてレンダリングされる。同じレンダ

リングルールで、 S_6 4606が S_P になり、4606のすべてが S_c に指定される。

スパン S_6 4606を S_4 4604と同様に処理するが、そこで x_{6L} 4634から x_{7L} 4636までのその第1ポーションが透明スパン S_5 4605と組み合わせられる。 x_{7L} 4635から x_{8L} 4636までの第2のポーションを透明スパン S_7 4607と組み合わせる。そして、 x_{8L} 4636の右の第3のポーションは隠れているため、 S_7 4607が S_P になる。もう透明スパンはないので、 x_{8L} 4636から右まで、前のセクションで説明したようにスパンを処理する。

バージョン2：深さ可視性が変化するところでラスタラインを片に分割する

この方法のこのバージョンでは、不透明か透明かを問わず、左から右へ最前スパンを追う。不透明スパンを左から右へ、可視透明スパンが始まるx座標までレンダリングする時、不透明スパンのレンダリングは一時的に中断され（本質的に、このレンダリングプロセスは1スタックにプッシュされる）、レンダリングは透明スパンにシフトする。最前透明スパンは次のいずれかまで左から右へレンダリングされる：1）透明スパンが、不透明スパンのみかスパンがまったく存在しないx座標で終わる、または2）最前透明スパンが不透明スパンに閉塞される。この時、レンダリングは、一時的に中断した不透明スパンのレンダリングに戻る（レンダリングプロセスのスタックがポップされる）。

透明スパンをレンダリングすると、これらをスパンメモリ408から削除し、方法の後で再レンダリングされないようにする。透明スパンは可視から隠れへトランジションでき、方法がさらに右までレンダリングすると再び可視になるため、既にレンダリングした可

視部分を削除して再レンダリングされないようにする必要がある。これは、スパ

ンメモリ408の透明スパンに上書きして、その左エンドポイントを変更することによって行う。こうして、透明スパンをレンダリングする時、その右エンドポイントまでレンダリングされると、完全に削除される。透明スパンが部分的にレンダリングされる時、レンダリングした部分のみ削除される。

可視透明スパンを1つしか持たないラスタラインのポーションでは、スタック指向のアプローチでまず透明スパンをレンダリングしてから不透明スパンをレンダリングする。可視透明スパンを2つ（他方の前に1つ）持つラスタラインのポーションでは、スタック指向アプローチで最前透明スパンを先にレンダリングしてから、フロントの透明スパンから2番目、そして最後に不透明スパンをレンダリングする。レンダリングプロセスのスタックがオーバーフローしない限り、この方法はあらゆる深さの可視透明スパンをレンダリングできる。

ラスタラインは、可視性深さが変化するx座標位置の片に分割されていると考えることができる（1から2、2から3、2から1等）。これらの片の中で、可視スパンを前から後ろの順でレンダリングする。前から後ろへレンダリングすることで、ピクセルカラー値を正しくブレンドできる。これは、ランダム順に幾何形状をレンダリングする先行技術方法（zバッファ技術等）とは対照的である。

図46に示すスパンを例に用いると、この方法のこのバージョンは、 S_0 4600と S_1 4601の可視ポーションを x_{2L} 4626までレンダリングする。この方法のこの時点で、 S_c は、 x_{2L} 4626と x_{3L} 4628の間の S_1 4601のポーションで、これはレンダリングプロセススタックにプッシュされる。次に、透明スパン S_2 4602、 S_5 4605および S_7 4607を、 x_{2L}

4626から x_{8L} 4636までレンダリングする。これらのスパンをレンダリングする時、 S_2 4602を削除し、 S_5 4605を削除し、 S_7 4607の部分をその左エンドポイントを (x_{8L}, z_{temp1}) 4640に変更することで削除する。次に、レンダリングプロセススタックがポップされ、 S_c が x_{2L} 4626と x_{3L} 4628の間の S_1 4601の部分に復元される。ここで、不透明スパンのレンダリングが再開し、スパン S_1 4601、 S_3 4603、 S_4 4604および S_6 4606の可視ポーションが x_{2L} 4626から x_{8L} 4636までレ

ンダリングされる。次の S_c を S_0 4 6 0 8 のすべてにセットし、不透明スパンのバランスをレンダリングする。 S_0 4 6 0 6 の残りの片は隠れているため、この方法はスパンメモリ 4 0 8 から 4 6 0 6 を読み出すことを避ける。

バージョン 3 : 複数パスの実行

この方法のこのバージョンでは、複数パス戦略を用いる。第 1 パスで、本書で前に説明した方法で不透明幾何形状をすべてレンダリングする。不透明スパンのレンダリング終了時、不透明スパンの有効フラッグ 8 1 6 はすべて切れている。そして、透明スパンをレンダリングする第 2 パスで、不透明スパンの有効フラッグ 8 1 6 をすべて入れ、各透明スパンを順次テストして、不透明スパンに閉塞されているか否か決定し、その可視ポーションをレンダリングする。

考えられるオプションとして、不透明スパンをレンダリングする時、照会を行って、完全に不透明スパンの背後にある透明スパンをすべて見つけ、そのような透明スパンの有効フラッグ 8 1 6 をデアサーションする。これによって、処理して隠れている透明スパンの数が減るが、別の照会を追加することで各不透明スパンが必要とする処理が増える。このオーバーヘッドを減らすため、隣接する不透明スパンの各セットの後で透明スパンを無効化する照会を実行するこ

とができる。そして、隣接するスパンセットのレンダリングが終了したら、そのセットのスパンのいずれかの可視ポーションの最大 z 値を照会の最小 z 値として使って、隠れた透明スパンを探す。このオプションは、次のパラグラフで説明する例で想定している。このオプションでは、1 つのスパンを 1 スパンセットと考え、閉塞透明スパンとして用いることができる。

図 4 6 に示すスパンを例として用いると、この方法のこのバージョンの第 1 パスでは、 S_0 4 6 0 0、 S_1 4 6 0 1、 S_3 4 6 0 3 および S_4 4 6 0 4 の可視ポーションをレンダリングする。この方法のこの時点で、隣接するスパンセットの、レンダリングは終了しており、そのセットによって隠された透明スパンは、透明スパンが完全に x_{1L} 4 6 2 4 から x_{6L} 4 6 3 4 の x 座標範囲内にあり、その両方のエンドポイントの z 値が z_{temp2} 4 6 4 2 より大きいことを見つける照会によって無効化される。次に、 S_0 4 6 0 6 の可視部分をレンダリングする。

そして、 S_8 4608と S_{10} 4610の隣接するスパンセットをレンダリングして、照会を行い、そのセットの背後にある透明スパンを無効化する。同様に、 S_{11} 4611と S_{12} 4612をレンダリングして、同じ種類の照会を行う。ラストラインのもう一方の不透明スパンは正常に処理する。第2パスの冒頭で、不透明スパンの有効フラッグ816をすべて表明し、透明スパンを左から右へ順次処理する。例では、 S_2 4602をスパンメモリ408から読み取り、4602についてSOT照会を行い、閉塞するスパンが見つからないため、レンダリングする。スパン S_6 4606を同様に処理する。 S_7 4607については、そのSOT照会で S_8 4608が見つかるため、 S_7 4607は x_{8L} 4636までのみレンダリングする。他の透明スパンが残っていないため、このラストラインは完了している。

この方法のこのバージョンの任意の変更として、スパンメモリ408とは別のメモリに透明スパンを保存するものがある。そして、第2パスで各透明スパンをこの別のメモリから読み取り、これについてSOT照会を行い、その可視部分をレンダリングする。

透明性の追加考察

多角形カラー化はしばしばテクスチャマップで記述される。テクスチャマップには、100%透明（完全にクリア）なピクセルを持つと定義されるものがある。このようなテクスチャマップの典型的な使用例は、矩形の「ビルボード形」多角形（ビルボード多角形は自動的に回転してビューイングポイントを向く）にマップされた木の画像で、ここで矩形の角はクリアで、木の画像内に穴が発生することがある。このようなテクスチャマップを持つ多角形は、そのスパンを透明スパンとして処理する必要がある。これらがその後ろの幾何形状のすべてを完全には閉塞していないためである。故に、多角形が透明ピクセルのテクスチャマップを使っている時は、その多角形のスパンは透明と想定し、テクスチャマップがその各ピクセルの不透明係数を決定する。

台形スパン（または四辺形スパン）を使用する際、スパン上部かスパン下部のいずれかを有するが両方は持たない多角形の角と辺を含むスパンは、透明スパン

として処理できる。これが可能なのは、この種のスパンは、透明スパンのようにより遠いスパンを完全には閉塞しないためである。故に、本書の透明性に関するこのセクションを使って、ラスタラインの縦高さを部分的にのみカバーするスパンを処理する。

四辺形で表すスパン

スパンを台形で表すことには、 $x-z$ 面（正確には面ではなく、1 ラスタラインの厚みを持つ）に4ポイントの位置を必要とするた

め、メモリビットを多く必要とするという不都合がある。これは、 z 値がそれぞれ32ビットの場合に特に当てはまる。

メモリ要件を減らすため、スパンのエンドポイント位置を2つの x 値と1つの z 値で近似させることができる。本質的に、エンドポイントを記述するこの2つのポイントは同じ z 座標を持つが、 x 座標は異なる。そのため、座標を保存するフィールドの数が8から6に減る。これによって、前述のアンチエイリアシング法に少量のエラーが導入されるが、エラーは $x-z$ 面にほぼ平行な多角形にとってのみ重要である。このような多角形では、ルック方向は多角形の面の接線に非常に近い。しかしながら、大半のアプリケーションで、これら多角形のビューイングプレーン102への投影は変化していないため、このエラーは許容可能である。

図47に、台形スパン S_1 4701、 S_3 4703および S_4 4704を点線で示し、これらスパンの四辺形近似、 S'_1 4711、 S'_3 4713および S'_4 4714を実線で示す。この図で、四辺形スパンのエンドポイントは常に水平ラインセグメントである。すべての台形スパンをその台形近似で置き換えている点を除き、図48は図40に類似である。台形スパンの一部は折り重なっているように見えることに注意する。台形スパンは、台形スパンについて本書で説明したのと類似の方法で処理できる。

カレント多角形メモリ406をなくしソートメモリ404から読み取る

バケット・ソート・プロセッサ504に「スマートさ」を追加することで、カレント多角形メモリ406とカレント多角形プロセッサ506をなくすことがで

きる。図49に、直接スパン生成4900を持つスパン・ソーティング・レンダリング・パイプラインを示すが、ここではバケット・ソート・メモリとカレント多角形メモリ

406からのデータの処理1600が（スパン・ソーティング・レンダリング・パイプライン400と比較して）除去されており、バケット・ソート・メモリ404はソートメモリ4904に変更されている。パイプライン4900の多角形処理1400部分で多角形を生成する時、そのパラメータをソートメモリ4904に書き込む。カレントラスタラインに影響する多角形の記録は、多角形パラメータを別のメモリ（カレント多角形メモリ406）にコピーするのではなく、ポイントのリストをソートメモリ4904内に保持することによって行う。これは、直接スパン生成5000を持つスパン・ソーティング・レンダラー・アーキテクチャに示すように、多角形をバケット・ソート・プロセッサ504から直接、高速でスパン生成プロセッサ508に送ることで達成する。ソートメモリ4904により高速でアクセスするため、これには、ソートメモリ4904に低速メモリではなく高速メモリが必要になる可能性があるという不都合がある。

この代替え案を用いる決定は、1つの大型高速メモリ（ソートメモリ4904）対1つの小型高速メモリ（カレント多角形メモリ406）と2つの大型低速メモリ（バケット・ソート・メモリ404）の両方の内いずれかというエンジニアリング上の決定に基づかなければならない。しかしながら、すべての多角形が特定のラスタラインに影響するような極端なケースでは、カレント多角形メモリ406はバケット・ソート・メモリ404と同じくらい大型である必要があるため、別にカレント多角形メモリ406を用いる場合でも大型高速メモリの必要があるだろう。

導関数を用いる多角形の代替え表現

考えられる改良方法として、パラメータの代替えセットを多角形とスパンに用いて、導関数を追加使用することができる。例えば、

図51に示すように、三角形は次のパラメータで表すことができる：1) 1つの

頂点5102の(x、y、z)、2) 第2の頂点5104の(x、y)、3) 第3の頂点5106のy、4) 図52に示す各頂点5202、5204、5206のそれぞれの(x、y、z)という一般的な表現ではなく、三角形の3辺5108、5110、5112の3つのそれぞれの $\delta x / \delta y$ 。いずれの場合も、三角形のパラメータ数は9である。この改良方法は、辺導関数5108、5110、5112が両方の表現(すなわち、図51と図52)のエッジウォーキングに必要で、計算・保存しなければならないため重要である。故に、より少ない頂点座標(図51のような)を用いるとメモリを節約できる。

サブスパンのオンザフライ生成

図38に示すように、複数のセグメントスパンをラスタライン内で用いてアンチエイリアシングを強化することができる。1つのラスタラインについて複数のセグメントスパンを持つことによる重要な不都合によって次のいずれかが必要になる: 1) 各ラスタラインで実行する処理の対応する増加、2) スパンメモリ408を含むハードウェアの対応する増加、または3) 処理の増加とハードウェアの増加の組み合わせ。

台形スパンと四辺形スパンに考えられる改良方法として、空間データ構造でスパン全体のパラメータを保存することができ、スパンをデータ構造から読み取ると、サブスパンのセットがオンザフライで生成され、ここでこれらサブスパンがフルスパンの高さに縦に分布され、各サブスパンがセグメントスパンとなる。この例を図53に示すが、ここでは8個のサブスパンをスパン5300について示す。各サブスパンの左エンドポイント、 $x_L[0]$ 5302、 $x_L[1]$ 5304、 $x_L[2]$ 5306、 $x_L[3]$ 5308、 x_L

$[4]$ 5310、 $x_L[5]$ 5312、 $x_L[6]$ 5314および $x_L[7]$ 5316は、頂点の座標、辺導関数、頂点からのyオフセットを用いて生成する。右エンドポイント、 $x_R[0]$ 5322、 $x_R[1]$ 5324、 $x_R[2]$ 5326、 $x_R[3]$ 5328、 $x_R[4]$ 5330、 $x_R[5]$ 5332、 $x_R[6]$ 5334および $x_R[7]$ 5336は同様に生成する。サブスパンを使う場合、エンドポイントの位置はスパン・パラメータ・レジスタ3104に保存されるので、

これを拡張して追加情報を保存できるようにする必要がある。

サブスパンを使う場合、カレント・スパン・ポーシオンを新しいスパンと比較するルールがサブスパン毎に適用され、ラスタライン内の追加多角形辺情報を与え、これがアンチエイリアシングとなる。ここでも、リアルタイムパフォーマンスのため、空間データ構造をSMCCAMに保存することができる。あるスパンのポーシオンを、そのスパンのサブスパンのポーシオンが可視であるため可視と決定する時、これらサブスパンポーシオンをラスタライズ・プロセッサ 5 1 2 に送らなければならない。しかしながら、可視サブスパンポーシオンの右エンドポイントのみ送る必要がある。可視スパンポーシオン 5 4 0 0 の例を図 5 4 に示すが、この右エンドポイントを $x_R[0]$ 5 4 0 2、 $x_R[1]$ 5 4 0 4、 $x_R[2]$ 5 4 0 6、 $x_R[3]$ 5 4 0 8、 $x_R[4]$ 5 4 1 0、 $x_R[5]$ 5 4 1 2、 $x_R[6]$ 5 4 1 4 および $x_R[7]$ 5 4 1 6 として示す。左エンドポイントは、前に送った可視スパンポーシオンの右エンドポイントと等しいため、ラスタライズ・プロセッサ 5 1 2 に送る必要はない。故に、右エンドポイントセットのシーケンスを、ラスタラインの長さに沿って移動する「ウェーブ」と考えることができ、ここで「ウェーブフロント」は任意の形状を取ることができる。「穴」

ができて「ウェーブ状フロー」を乱さないよう特別に注意しなければならない。これは、最も最近送った右エンドポイントセット中の最左ポイントで可視サブスパンを見つけることで行う。図 5 4 の例では、次に可視のスパンポーシオンはポイント $x_R[3]$ 5 4 0 8 に位置しなければならない。

台形スパンの代替え表現

スパンメモリ 4 0 8 に保存したスパンの照会可能なパラメータの最小数は、1) スパン、 $x_L[r_{minx}]$ 5 5 0 2 の最左 x 値、2) スパン、 $x_R[r_{maxx}]$ 5 5 0 4 の最右 x 値、および 3) スパンのあらゆる角に位置できるような、スパン、 z_{min} 5 5 0 6 の最前 z 値である。これら 3 つのパラメータは、算術比較を実行できる SMCCAM 内のフィールドに保存しなければならない。しかしながら、これを生成し、スパンパラメータ、スパンのサブスパンエンドポイント、スパン内のサブスパンのあらゆるポイントの z 値を形成することが容易なことも重要で

ある。計算によって次の2種類の結果を生成する必要がある：1) スパン内またはその辺のあらゆるポイントでの z 値、および2) スパンの投影された辺を定義する x 値。図55に示す基準ポイント (x_{Ref} , z_{Ref}) 5508は、ラスタライン内の最上サブスパン y 値に (スパンがラスタラインのその部分に接触していなくとも) 位置することができる。この基準ポイント5508がわかっているならば、そのスパンの z 値は、どの位置にあってもこの基準ポイント5508とオリジナルの多角形の導関数、 $\delta z / \delta y$ 5510と $\delta z / \delta x$ 5512の x と y のオフセットから計算することができる。スパンの投影辺は同様に計算可能でなければならない、パラメータには多数の選択肢が可能で、考えられる選択肢には次のものが含まれる：1) 辺導関数 $\delta x_R / \delta y$ 5514と $\delta x_L / \delta y$ 5516、2) 最上サブスパン位置、 r_{Top} 551

8、3) 最下サブスパン位置、 r_{Bot} 5520、4) スパン、 $x_L [r_{Minx}]$ 5502の最右 x 値を含むサブスパン位置、 r_{Minx} 5522、および5) スパン、 $x_R [r_{Maxx}]$ 5504の最右 x 値を含むサブスパン位置、 r_{Maxx} 5524。

ラスタラインが多角形の角を含むなら、そのスパンは台形状を持たない。この例を図56に示すが、ここには中間 (上下ソートで) 角が含まれる。この例で結果として生じたスパンは5つの辺を持つため、台形にモデル化することができず、スパンは次の2つのスパンに分割される：1) 上台形5602、および2) 下台形5604。多角形の最上角または最下角が含まれる場合、スパンはディジェネレート台形としてモデル化し、ここで4つの角の内2つが同一である。

四辺形スパンに同様の代替え表現を用いることができる。

レンダリング順の保存

アプリケーションによっては、レンダリング順を保存しなければならない。典型的な例は、コンピュータのウィンドウ機能システムに用いる「ペインターズ・アルゴリズム」(Foley 参考文献、674頁参照) である。境界のある内側矩形のレンダリングでは、まず境界を充填矩形としてレンダリングしてから、内側矩形をレンダリングし、境界矩形のピクセルの大半に上書きする。これら2つの矩形のレンダリング順を逆転させると、内側矩形が完全に上書きされるので、領域

全体が境界矩形のカラーとなる。これら2つの矩形が、z座標の相違によって明確な配置を持っていれば、正しくレンダリングされる。しかしながら、z座標が指定されていないと（デフォルトはビューイングプレーン102の深さとなる）、順番を保存しなければならない。これは、バケット・ソート・メモリ514を持つことで可能で、カレント多角形プロセッサ506が、多角形プ

ロセッサ512から受け取る順番を維持する。照会プロセッサは、後に入力された多角形のスパンをスパンリストの後の位置に入れることでレンダリング順を保持する。そして、レンダリングが進むにつれ、z座標がタイの場合、リストの先のスパンを先にレンダリングし、同じz座標で後の幾何形状は隠れていると想定するため、レンダリング順が保存される。

方法と装置のオプションと考えられる強化法

シングルバッファVGAグラフィックス・カードへの追加

コンピュータのビデオ出力（IBM互換機のVGAやSVGAインターフェースなど）には一般に、ディスプレイ装置（すなわち、CRT）に送られるピクセル情報を保存するフレームバッファメモリが含まれる。本書に記述する3Dグラフィックス・レンダリング方法および／または装置をこのようなディスプレイシステムに追加することができる。CPU（またはDMAコントローラ）がレンダラーに幾何形状を書き込み、処理して、バケット・ソート・メモリ（ダブルバッファの必要あり）に保存する。幾何形状を書き込んでいる時、パイプラインの残りはコンピュータディスプレイにシンクロして、必要になる前にスパンを生成し、コンピュータのフレームバッファの前のフレームのデータに上書きする。故に、低コストだが高性能の3Dグラフィックスアクセラレータを、フレームバッファにメモリを追加することなくコンピュータに加えることができる。

特定多角形辺のアンチエイリアシングを切る

前述したアンチエイリアシング方法では、各多角形が独立したカラー値を持つと想定し、隣接するスパンが見つければ、両方のスパンを含むピクセルは、2つの多角形からブレンドしたカラーを持つ必要がある。しかしながら、辺を共有する多角形はしばしば辺に沿

って正確に同じカラー情報を持つ。例えば、Gouraudシェーディングを施した多角形で、同じ多角形辺の両端に同じカラーおよび面の正規情報を持つものにこれが発生する。このような辺では、2つの同じカラーがブレンドされるため、ピクセルのブレンドを無視できる。かかる多角形辺は、レンダラーに供給するとき、「スキップ・アンチエイリアシング」というタグを付けることができる。そして、もしこのような辺が隣接するスパンペアの一部に見つかった場合、その内の一方（例えば右）のピクセルカラー値をスキップすることによって、計算を節約する。タグが付くが、隣接するスパンペアの一部でない辺は、上記アンチエイリアシング方法によって処理する。先行技術のレンダラーは、隣接するスパンや多角形を検索しないため、このような節約の利点はない。すなわち、先行技術レンダラーは一般に各多角形が他のすべての多角形から独立していると想定している。タグ付けが行われない場合（インターフェースソフトウェアによる制約などにより）、スパン・ソーティング・レンダリング・パイプライン400は、2つの多角形に共有される辺が両多角形について同じカラー情報を持っている場合、多角形メッシュおよびストリップ内の辺について自動的にこのようなタグを生成することができる。

バケット・ソート・メモリ404に多角形ではなく辺ペアを保存

バケット・ソート・メモリ404に多角形を保存するのではなく、多角形の辺ペアを保存することができる。これによって多角形が水平にスライスされるため、多角形の各水平ストリップに多角形の角が入らない。故に、多角形が y_{START} でも y_{END} でもない角の y 座標（または近接するラスタライン境界）で水平にスライスされる。この多角形スライスにより、多角形角でいつ新しい辺に変更するかテストする必要がなくなるため、下流処理が容易になる。

フロントおよびリアのクリッピングプレーン

クリッピングプレーンを追加することで、ビューボリューム（無限ピラミッドの形状）を錐台に変更する。スパンをスパンメモリ408に書き込む直前に、これらのエンドポイントの z 値を、フロントおよびリアのクリッピングプレーンの z 位置と比較する。両エンドポイントがフロントのクリッピングプレーンより近

いか、リアのクリッピングプレーンより遠ければ、スパンを放棄する。1つのエンドポイントがリアのクリッピングプレーンより遠ければ、スパンのその端部を短縮して、そのエンドポイントがリアのクリッピングプレーンに来るようにする。1つのエンドポイントがフロントのクリッピングプレーンより近ければ、そのスパンの端部を短縮してそのエンドポイントがフロントのクリッピングプレーンに来るようにする。これには、スパン生成プロセッサ508か照会プロセッサ510のいずれかにコンパレータセットを追加する必要がある。

複数の任意クリッピングプレーン

任意のクリッピングプレーンをシンプルなフロントおよびリアクリッピングプレーンと同様に用いることができるが、この場合、スパンメモリ408にスパンを書き込む直前に、スパンとクリッピングプレーンのスパンの間のスパン・インタラクション・パラメータを生成して、これらを上記説明したように使ってスパン・インタラクション・タイプ(SIT)を決定する。そしてSITに応じてスパンを保持するか、クリップするか、あるいは放棄する。

これは、任意のクリッピングプレーンの位置を表すダミーのスパンをスパンメモリ408に追加することでも行える。そして上記のラスタライゼーション方法を変更して、レンダリングしたスパンすべてがこれらダミースパンの背後にあるようにする。

幾何形状キャッシュ

パイプラインの入力にメモリを加えて、幾何形状をレンダリングのために入力する時にこれを保存することができる。そして、後続シーンをレンダリングする時、この保存した(または「キャッシュした」)幾何形状を再レンダリングすることで(おそらく変更した翻訳マトリクスによって)、システムバスのサイクルをセーブすることができる。これによって、1秒あたりにホストコンピュータのバスがサポートできるより多くの幾何形状を処理できるレンダラーによって生じるボトルネックを解決することができる。これは、前回のレンダリングから形状を変更していない幾何形状にのみ適用できる。「スクイッシュ」なオブジェクトは一般に各シーンについてレンダラーに供給する必要がある。

スパンメモリ408の1ページのみ使用

スパンメモリ408を1ページのみ使用することが可能である。これは、有効スパンを上ワードに向かってシフトし、下ワードは無効のまま残すことで行う。そして、新しいスパンが生成されると、これらを下ワードに保存する。この戦略は、可動境界で2ページに分けた1つのメモリバンクと考えることができる。これが可能なのは、ラスタラインを処理する時、スパンを削除しながら、次のラスタラインを準備する同時プロセスでスパンを生成するためである。

代替えアプローチとして、固定ワード番号フィールド802を可変タグフィールドに変更することができる。各スパンについて一意の(ランダムでもよい)タグをこのタグフィールドに書き込み、次のものを指定する: 1) スパンがカレントラスタラインにあるか、次のラスタラインにあるか、および2) スパンスパンの一意の識別子。さらに、このタグを照会プロセスに含めて、カレントラスタラインのスパンを次のラスタラインのものと混合することができる。同じラスタラインのスパンが互いに対する順番を保持する限り、無

関連のラスタラインのスパンは、無視されていれば混合できる。これは、各ラスタラインに1つずつ、2つの有効フラッグ816セットを維持することで可能である。有効フラッグ816のセットをさらに追加することで、より多くのラスタラインのスパンを同時に保持することができる。

スパンメモリページの別個の読み取りバスと書き込みバス

SMCCAMセル3400または3500の回路を修正して、入力バスと出力バス両方を持たせることができる。これによって読み取りオペレーションと書き込みオペレーションを同時にできる。これは、前のセクションで説明したようにスパンメモリ408に1ページしかない場合に特に有益である。

境界ボックスが重なる時のみSIPとSITを計算

新しいスパンSNすべてについてSITを計算するのではなく、境界ボックスが重なる時のみより少ないハードウェアで順次計算する。これは回路を節約するため時間を犠牲にする。

SMCCAMメモリを節約するためラスタラインを小片に分割する

スパンメモリページ800のワード総数は、あるラスタラインで処理できるスパン数に制約を加えている。この制約は、ラスタラインを条件付きで半分（または必要ならそれ以上）に分割し、スパンデータを2回通過することで解決できる。この決定は、シーンをレンダリングする時オンザフライで、またはレンダリングが始まる前にアプリオリで行うことができる。

オーバフローRAMでソートオペレーション中にシフトアウトされたスパンをセーブする

ラスタラインを分割するのではなく、スパンメモリページ800に、最後のSMCCAMワード900からシフトアウトされたスパンを保存する「オーバフローRAM」を追加することができる。シ

フトアウトされたスパンはラスタラインの右から始まるので、xCLが最左シフトアウトスパンの左に達するまで、ラスタラインの左側は正常に処理できる。そして、ラスタライズプロセスによってスパンを削除（すなわち、無効化）しているので、シフトアウトスパンをスパンメモリページ800に再書き込みできる。

多角形ではなく基本立体のレンダリングを含める

多角形ではなく幾何学的基本立体（CSG基本立体など）を、スパン・ソーティング・レンダリング・パイプライン400で処理することができる。y_{START}およびy_{END}パラメータを計算でき（そして基本立体をバケット・ソート・メモリに保存でき）、境界ボックスのあるスパンを生成できればこれが可能である。境界ボックスの角をスパンパラメータとして用いて、SOT照会を使って境界ボックスの可視ポーションを見つける。特定の幾何学的基本立体セットには、スパン・インタラクション・タイプの別のセットが必要になる。

低コストバージョンでCPUをフロントエンドに使える

ホストCPU、または専用汎用プロセッサで多角形プロセッサ502のタスクを実行し、スクリーン座標の幾何形状（オブジェクト精密）をパイプラインの残りに送ることができる。同様に、このようなCPUでパイプライン400のフロントエンドをより多く実行することができる。例えば、CPUとその汎用メモリは、多角形プロセッサ502、バケット・ソート・プロセッサ504、バケット

・ソート・メモリ404のタスクを実行し、現在のラスタラインでスタートする多角形をパイプラインの残りに送ることができる。

代替えソーティングメカニズムでアップカウンティングタグを使える

ワード番号フィールド802を条件付きアップカウンタに置き換

え、SMCCAMからのシフティングをなくすことができる。新しいスパンをSMCCAMに書き込む時、ソートしたリストのどのスパンが新しいスパンより後かを照会で決定する。このような「後の」スパンは自分のアップカウンタを増分し、シフティングは実行されない。新しいスパンのアップカウンタを、アップカウントする前に「後の」スパンのリストの最小カウンタ番号に割り当てる。

このアプローチは、「保存リストの最初のスパンを取得」が、スパンメモリ408の「最小カウンタ値を見つける」になることを意味する。故に、ソーティング（すなわちシフティング）が新しい検索タイプと置き換わる。これを速くするには、ハードウェアにアップカウンタセットに関連する「最小発見」機能を持たせる必要がある。また、この「最小発見」をビットシリアル方法で実行することもできる。

ジェネリックCAMを使う

ジェネリックの先行技術CAMを使うことができるが、マグニチュード比較はビットシーケンシャルになる。これによって処理は非常に遅くなるが、ハードウェアのコストをやや節約できる。この代替え案ではアップカウンティングタグを前のセクションのように用いることになる。

1つの算術コンパレータに複数のレジスタを使う

SMCCAMの上記説明では、メモリレジスタと算術コンパレータに1対1の対応を想定している。代替えアプローチとして、同じワードのすべてのメモリレジスタが1つの算術コンパレータを共有するものがある。これによって回路は節約できるが処理は遅くなる。しかしながら、すべてのフィールドがすべての照会オペレーションに参加するわけではないので、パフォーマンスロスフィールド総数には比例しない。

別の代替アプローチとして、ワード間、またはワードとフィールド間でコンパレータを共有することができる。このアプローチでハードウェアの量はさらに減るが、パフォーマンスもさらに落ちる。このような共有コンパレータアプローチのいずれかを用いる場合、最も密度が高いので標準型6トランジスタRAMセルを使わなければならない。故に、検索およびソーティングハードウェアは複数ワードメモリセットで構成することができ、このような各メモリはそれに関連するコンパレータを1以上持つことになる。

RAMベースソーティングプロセッサを使う

あらゆる種類のCAMの使用の代替案として、高速リンクのリストソーティングプロセッサを用いることができる。高速メモリを利用することで、スパンを前述(SMCCAMを用いる)と同じ順番で線形リストにソートすることができる。しかし、スパンを頻繁に追加・削除するため、このアプローチは非常に遅くなる。利点は、容易に入手可能な高速の市販RAMを使えることである。

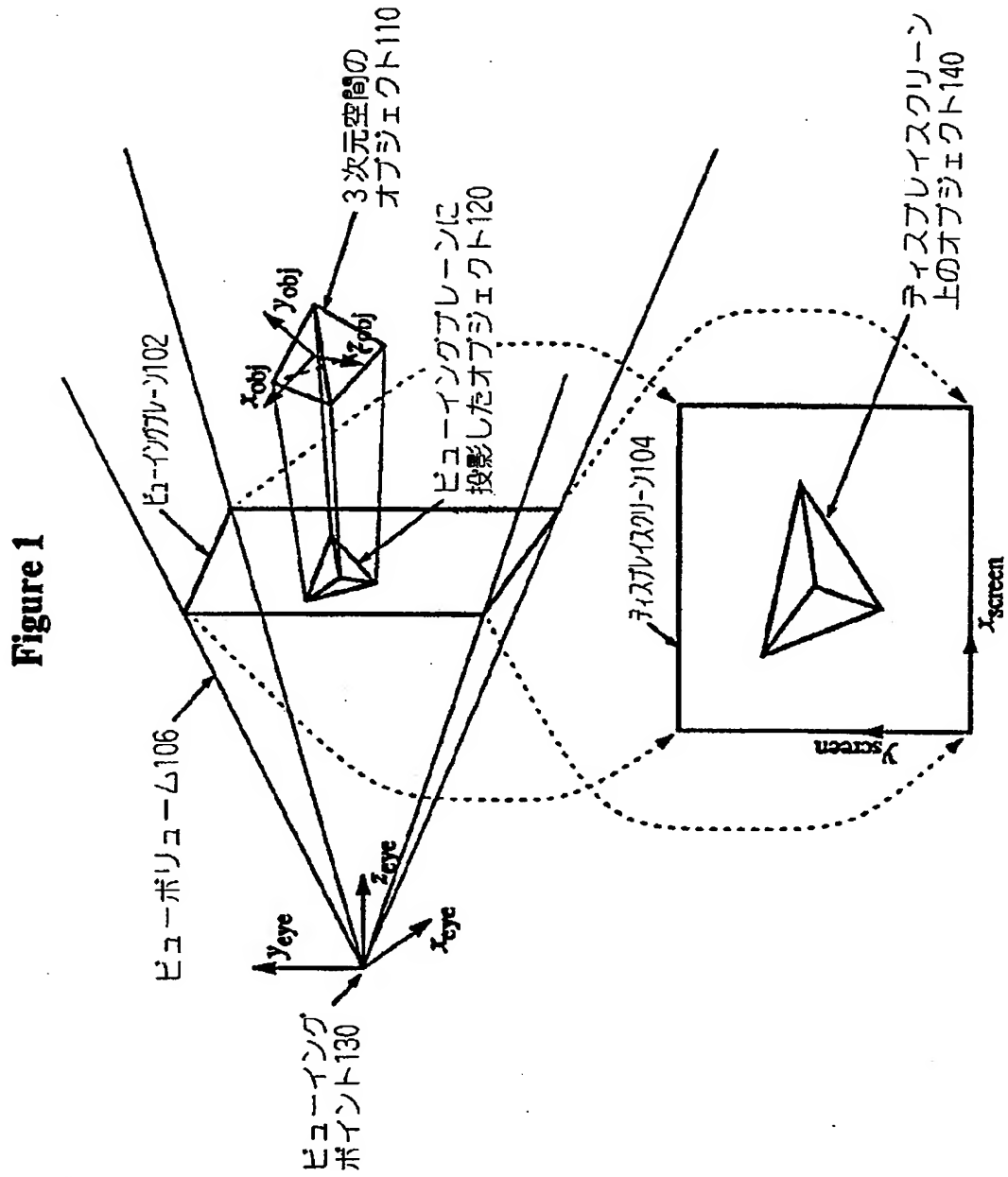
SOT照会オペレーションを助けるため、2次元データ構造を使ってxおよびz次元のスパンを同時にソートすることができる。このような検索およびソーティング構造は、Kurt Mehlhorn による「Data Structures and Algorithms 3: Multi-dimensional Searching and Computational Geometry」、24から55ページ、Brauer, W. 他監修による「Monographs on Theoretical Computer Science」のEATCS シリーズ、ISBN 0-387-13642-8、Springer-Verlag 社、ニューヨーク、1985年、に記載されている。

シャドー処理

シャドー計算は隠面消去と本質的に同じであるため、スパン・ソーティング・レンダラー500をシャドー計算に用いることができる。これは、複数のスパン・ソーティング・レンダラー500を平

行して用いるか、または1つのスパン・ソーティング・レンダラー500を時分割することによって可能である。

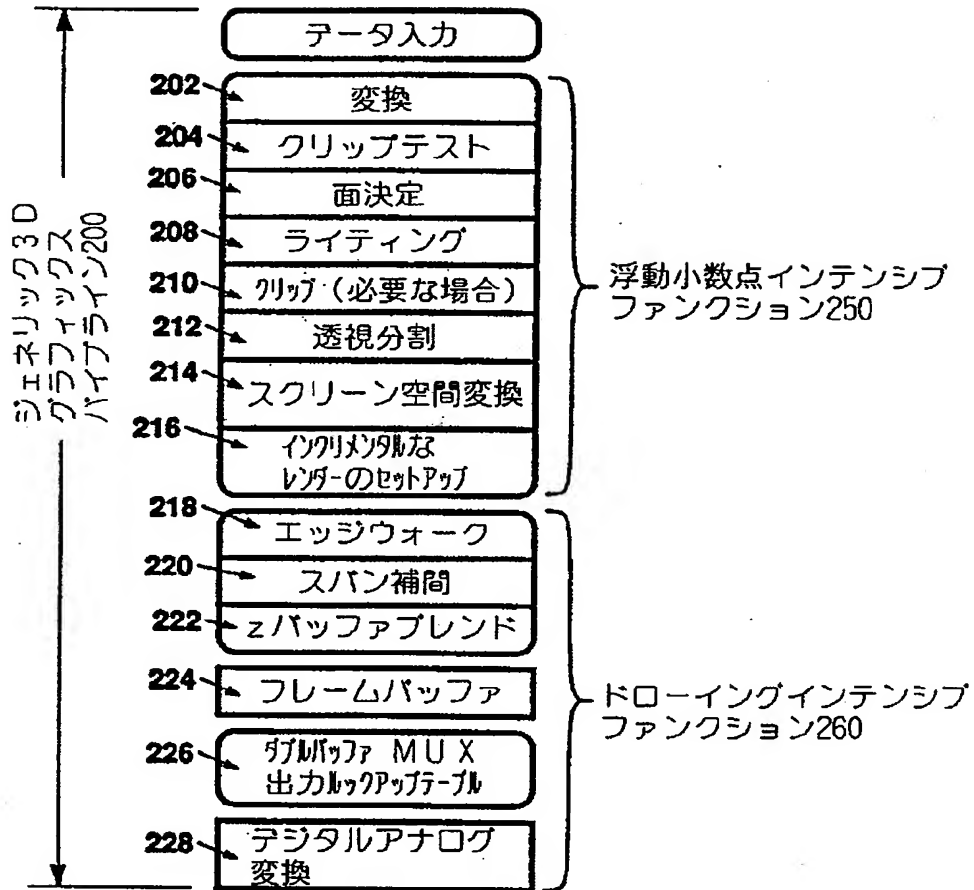
【図 1】



【図2】

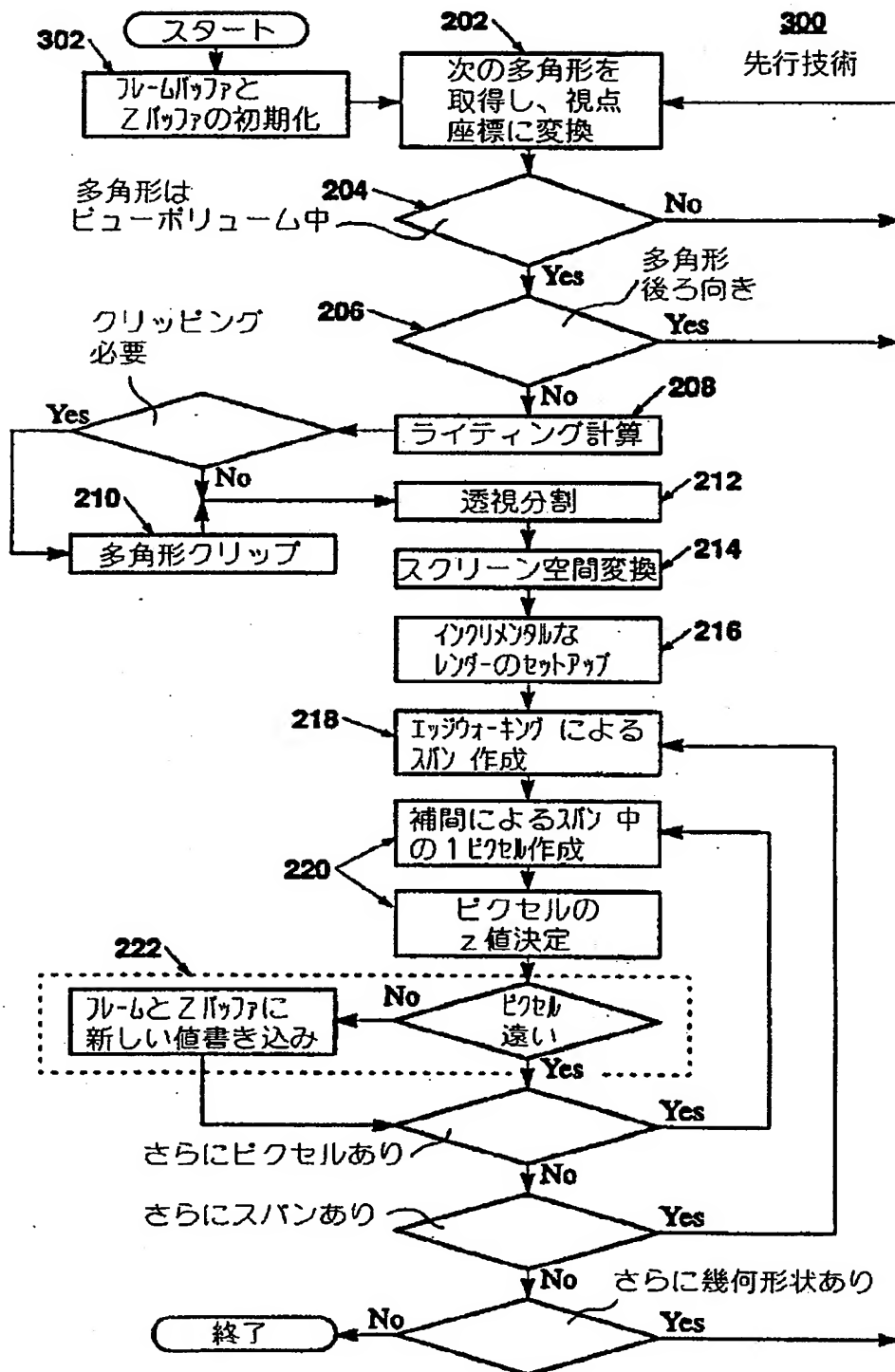
Figure 2

200
先行技術



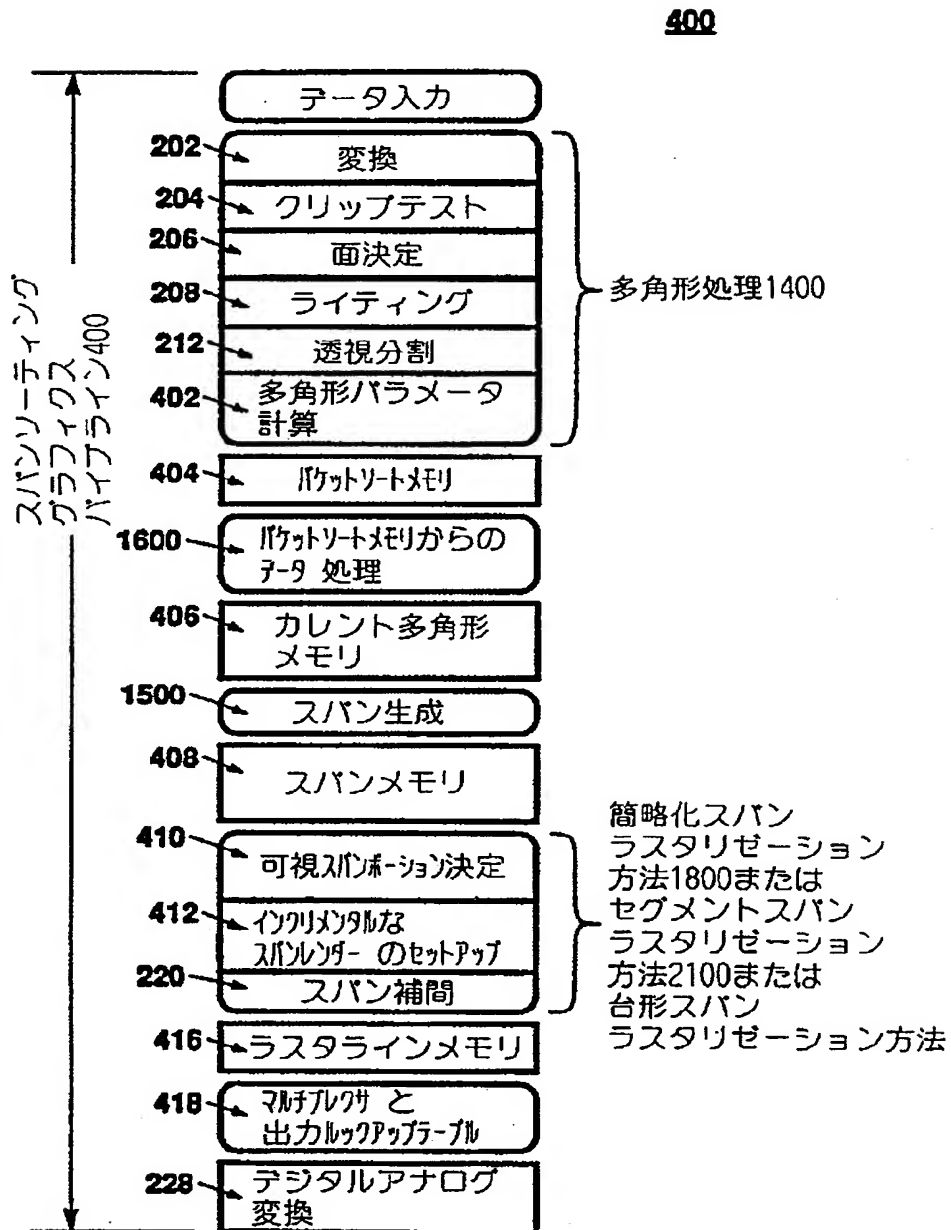
【図3】

Figure 3



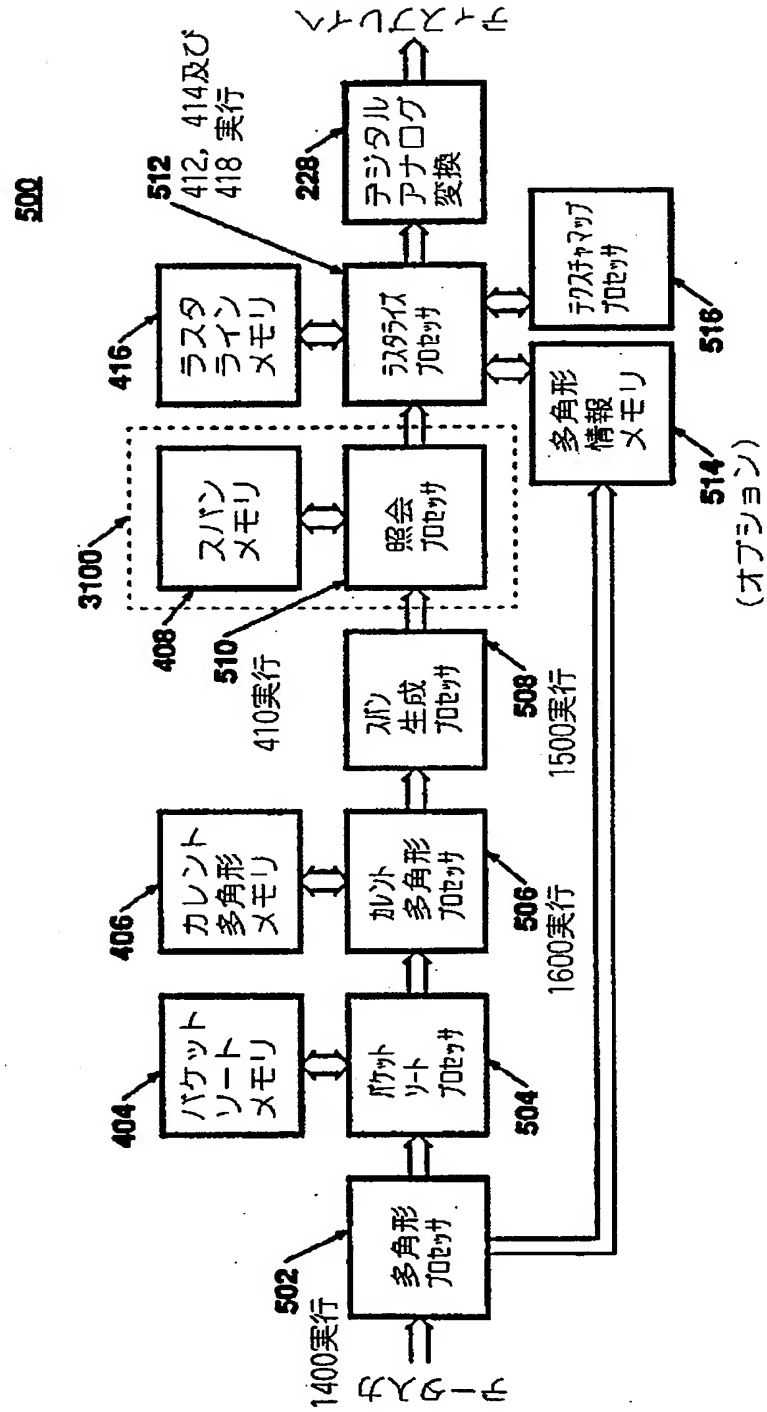
【図4】

Figure 4



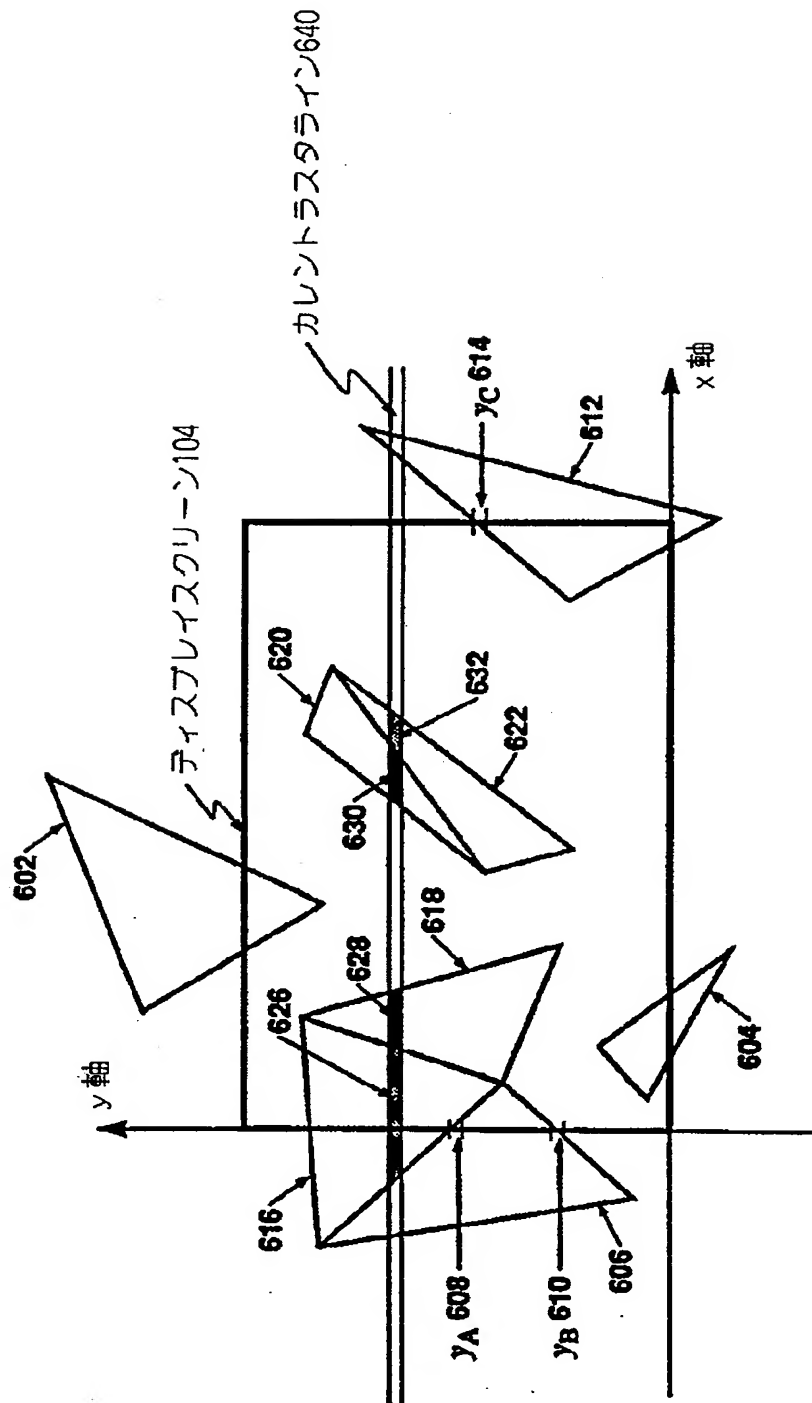
【図5】

Figure 5



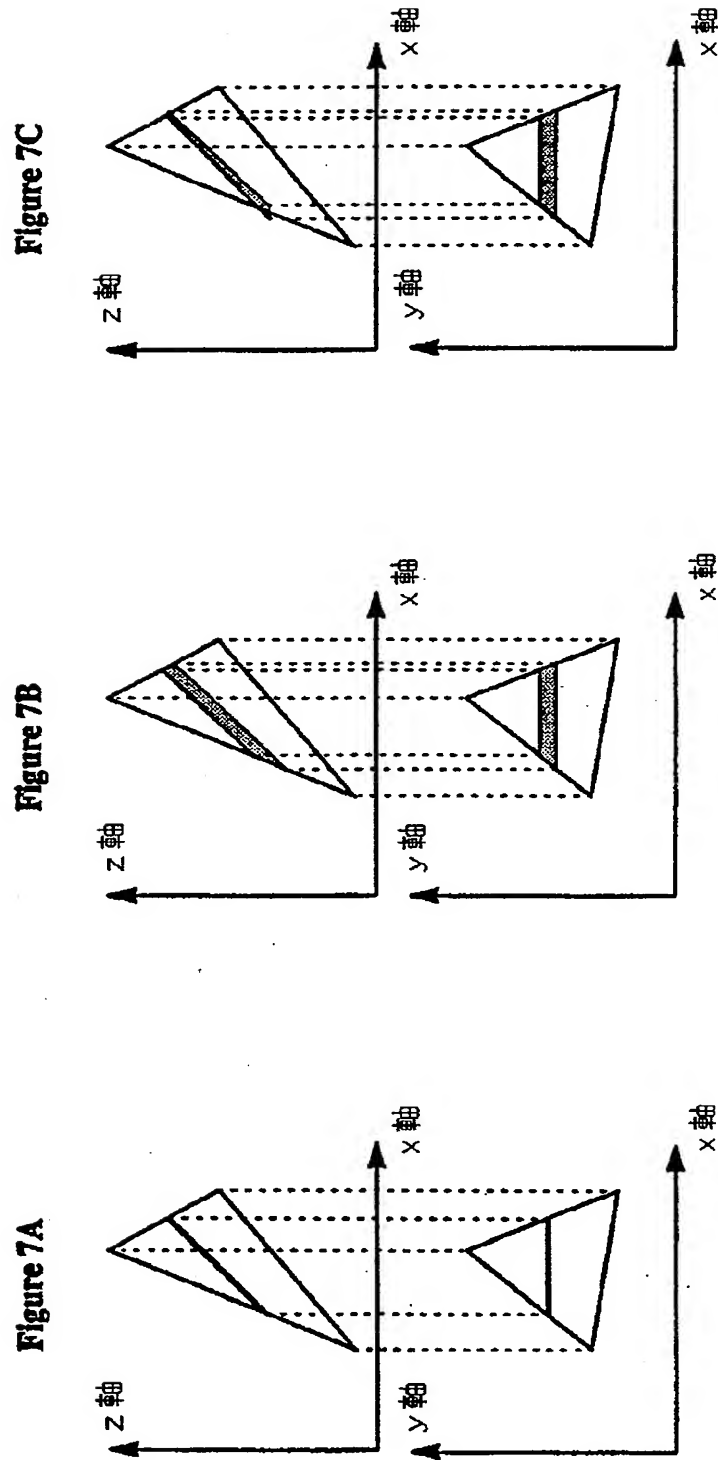
【図 6】

Figure 6



【图 7】

Figure 7



【図 8】

Figure 8

800

有効フラグワード Fv 830

802

照会
フラグ
ワード
FQ 834

1つの
SMC
CAM
ワード
900

$w = W - 1$	$x_{(w-1)L}$	$z_{(w-1)L}$	$x_{(w-1)R}$	$z_{(w-1)R}$	$d_{(w-1)}$	$i_{(w-1)}$	$F_{(w-1)V}$	$F_{(w-1)Q}$
$w = W - 2$	$x_{(w-2)L}$	$z_{(w-2)L}$	$x_{(w-2)R}$	$z_{(w-2)R}$	$d_{(w-2)}$	$i_{(w-2)}$	$F_{(w-2)V}$	$F_{(w-2)Q}$
$w = W - 3$	$x_{(w-3)L}$	$z_{(w-3)L}$	$x_{(w-3)R}$	$z_{(w-3)R}$	$d_{(w-3)}$	$i_{(w-3)}$	$F_{(w-3)V}$	$F_{(w-3)Q}$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
$w = m + 1$	$x_{(m+1)L}$	$z_{(m+1)L}$	$x_{(m+1)R}$	$z_{(m+1)R}$	$d_{(m+1)}$	$i_{(m+1)}$	$F_{(m+1)V}$	$F_{(m+1)Q}$
$w = m$	x_{mL}	z_{mL}	x_{mR}	z_{mR}	d_m	i_m	F_{mV}	F_{mQ}
$w = m - 1$	$x_{(m-1)L}$	$z_{(m-1)L}$	$x_{(m-1)R}$	$z_{(m-1)R}$	$d_{(m-1)}$	$i_{(m-1)}$	$F_{(m-1)V}$	$F_{(m-1)Q}$
$w = m - 2$	$x_{(m-2)L}$	$z_{(m-2)L}$	$x_{(m-2)R}$	$z_{(m-2)R}$	$d_{(m-2)}$	$i_{(m-2)}$	$F_{(m-2)V}$	$F_{(m-2)Q}$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
$w = n + 1$	$x_{(n+1)L}$	$z_{(n+1)L}$	$x_{(n+1)R}$	$z_{(n+1)R}$	$d_{(n+1)}$	$i_{(n+1)}$	$F_{(n+1)V}$	$F_{(n+1)Q}$
$w = n$	x_{nL}	z_{nL}	x_{nR}	z_{nR}	d_n	i_n	F_{nV}	F_{nQ}
$w = n - 1$	$x_{(n-1)L}$	$z_{(n-1)L}$	$x_{(n-1)R}$	$z_{(n-1)R}$	$d_{(n-1)}$	$i_{(n-1)}$	$F_{(n-1)V}$	$F_{(n-1)Q}$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
$w = 3$	x_{3L}	z_{3L}	x_{3R}	z_{3R}	d_3	i_3	F_{3V}	F_{3Q}
$w = 2$	x_{2L}	z_{2L}	x_{2R}	z_{2R}	d_2	i_2	F_{2V}	F_{2Q}
$w = 1$	x_{1L}	z_{1L}	x_{1R}	z_{1R}	d_1	i_1	F_{1V}	F_{1Q}
$w = 0$	x_{0L}	z_{0L}	x_{0R}	z_{0R}	d_0	i_0	F_{0V}	F_{0Q}

照会
フラグ
818

有効
フラグ
816

情報
814

導関数
812

右 Z
810

右 X
808

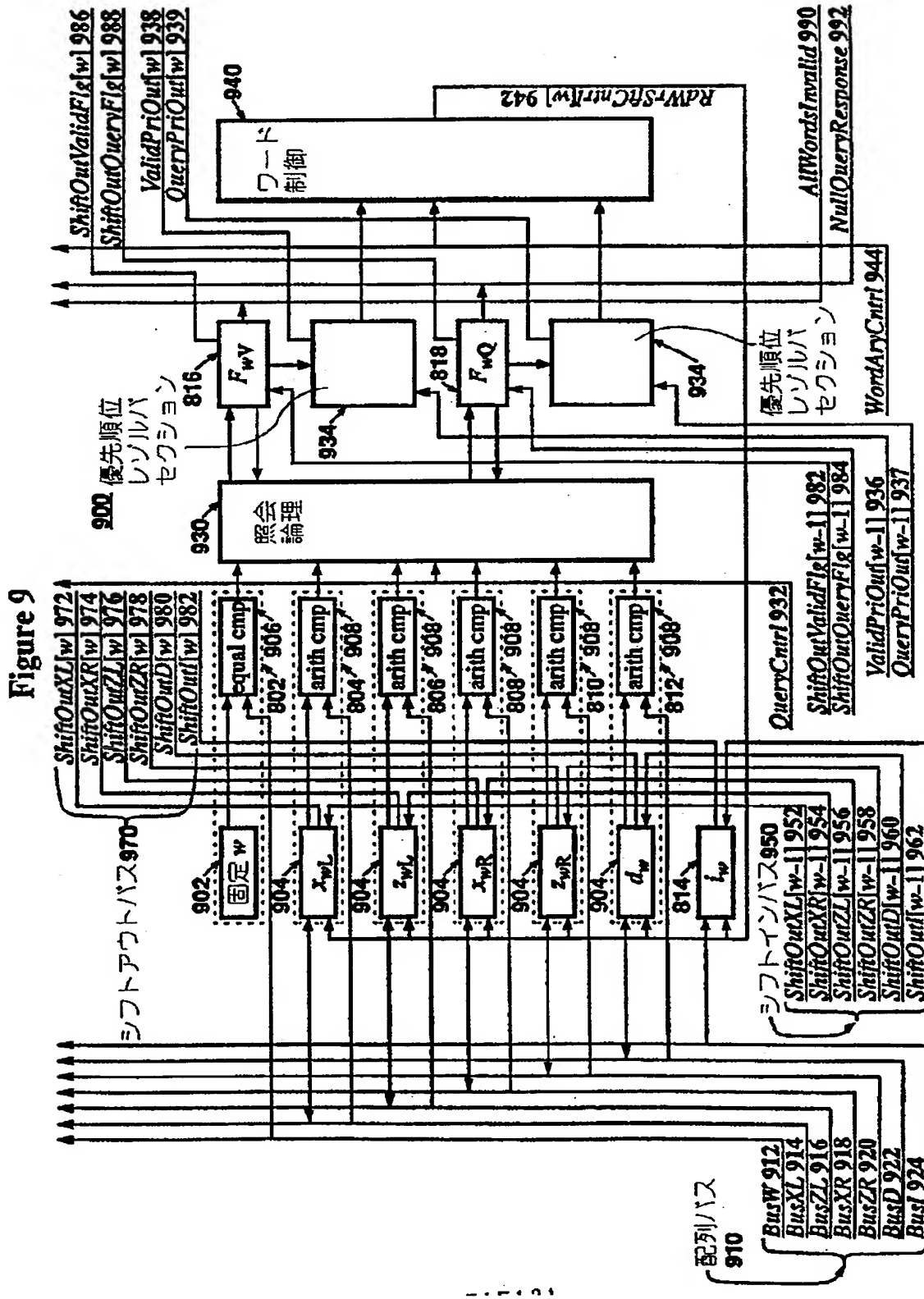
左 Z
806

左 X
804

ワード
番号
802

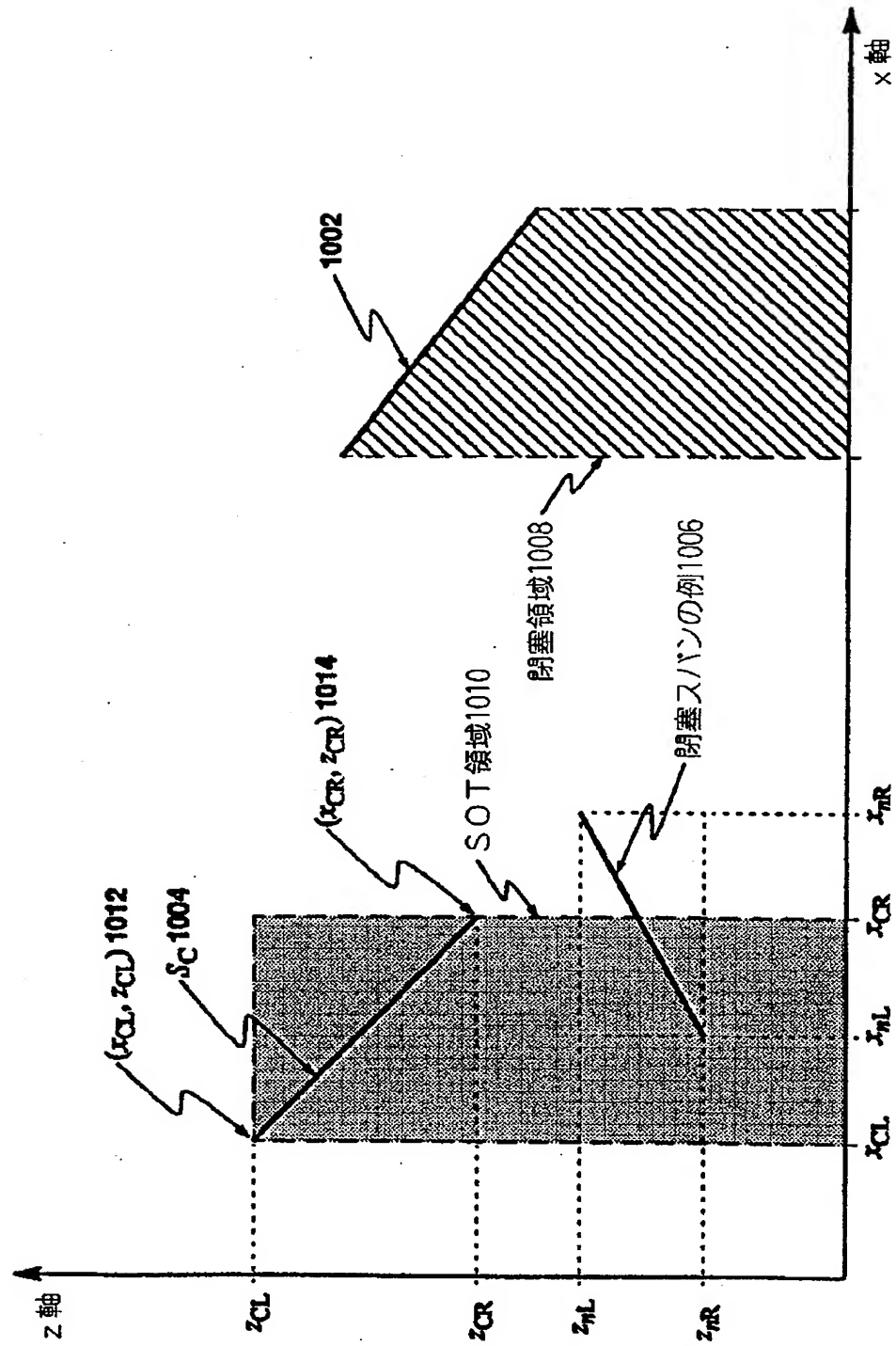
1つの
SMC-
CAM
ワード
900

【図 9】



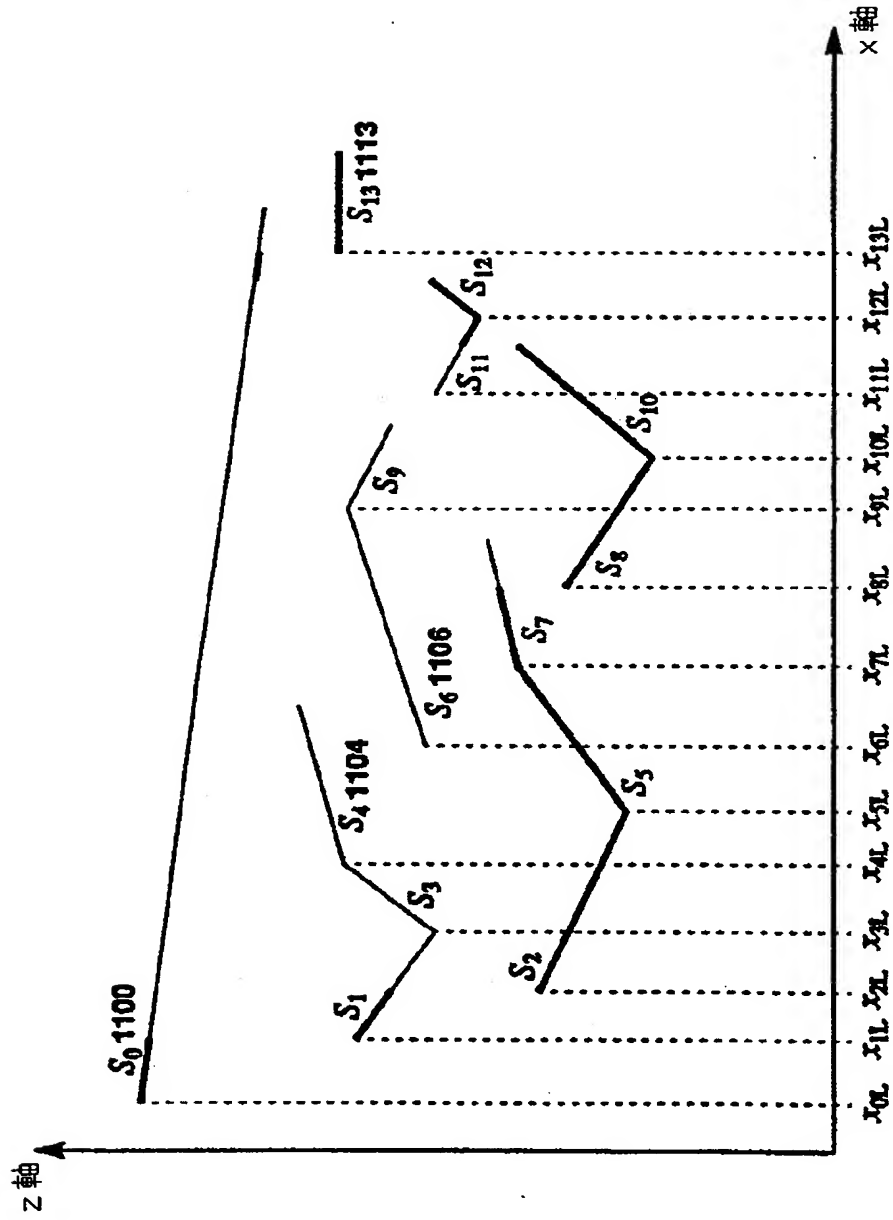
【図10】

Figure 10



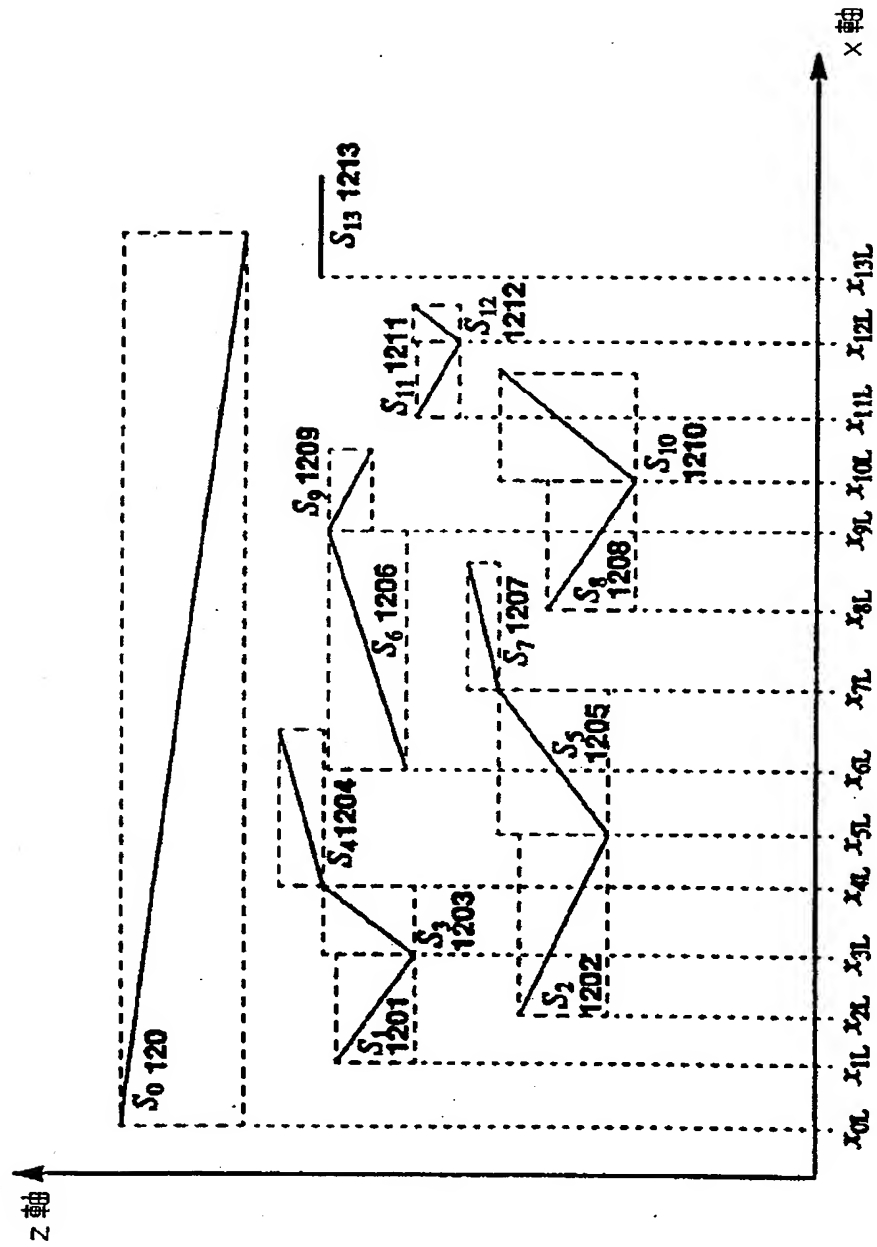
【図 1 1】

Figure 11



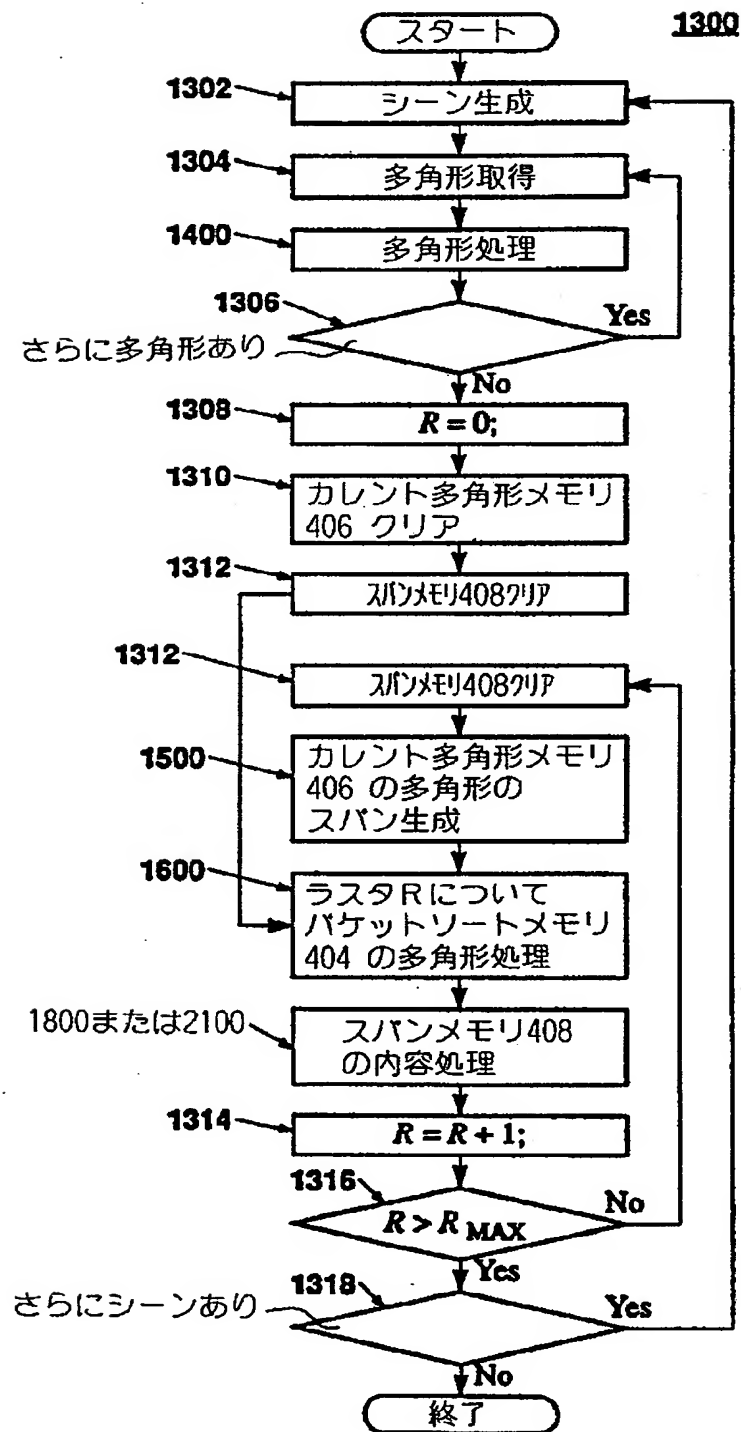
【図 1 2】

Figure 12



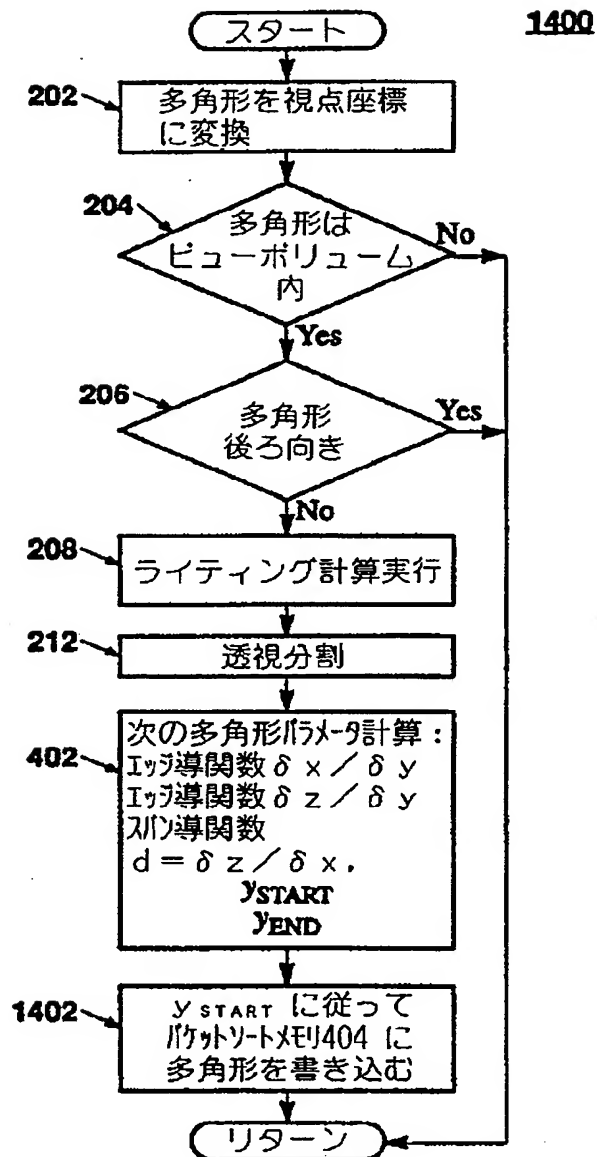
【図13】

Figure 13



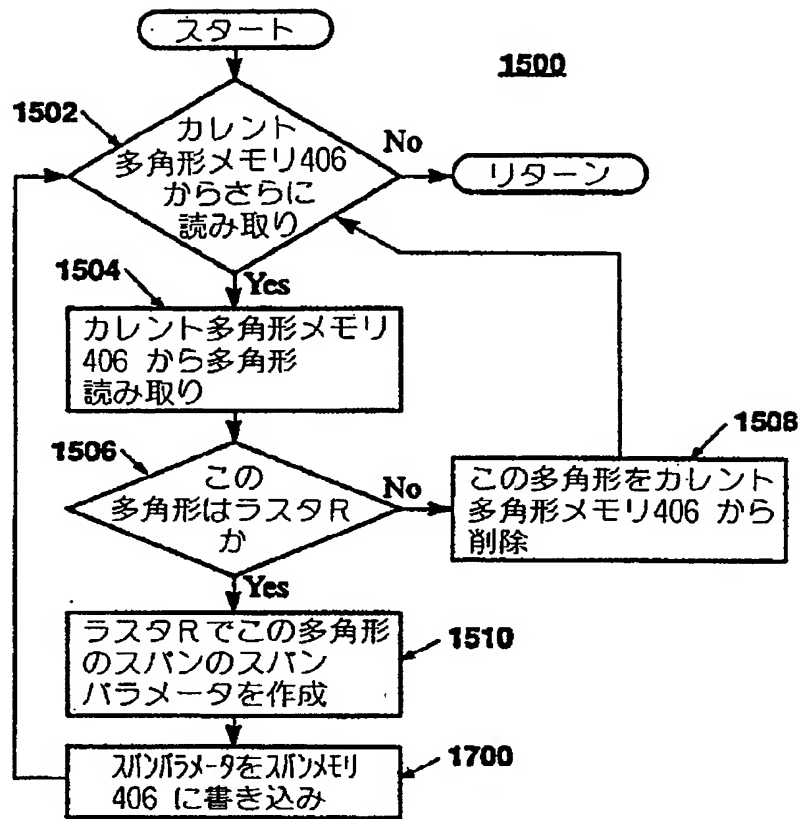
【図 1 4】

Figure 14



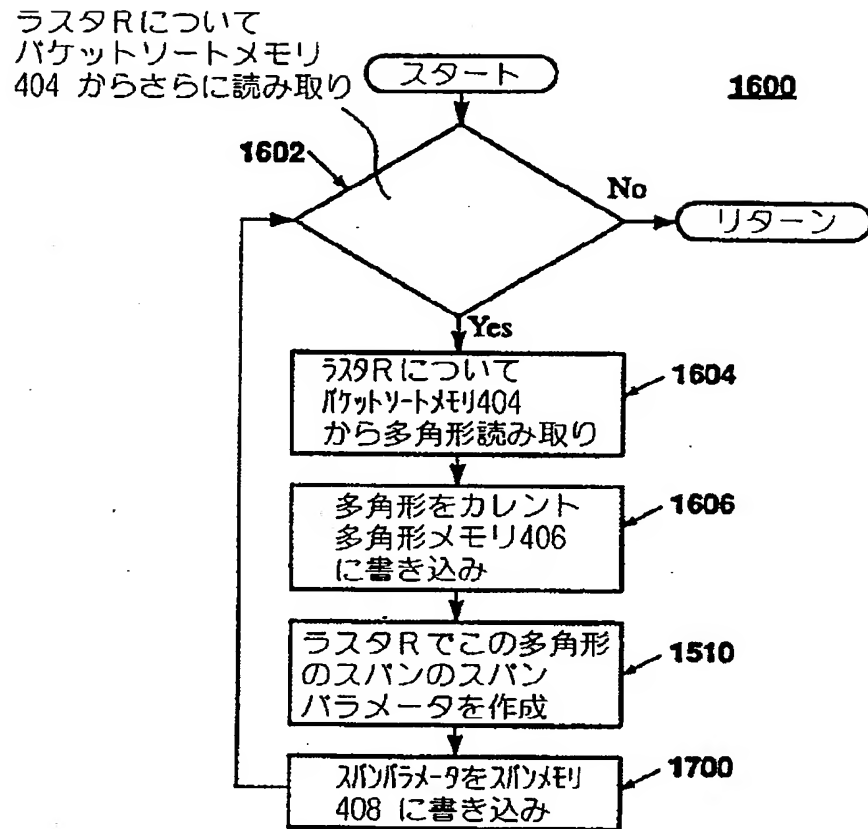
【図15】

Figure 15



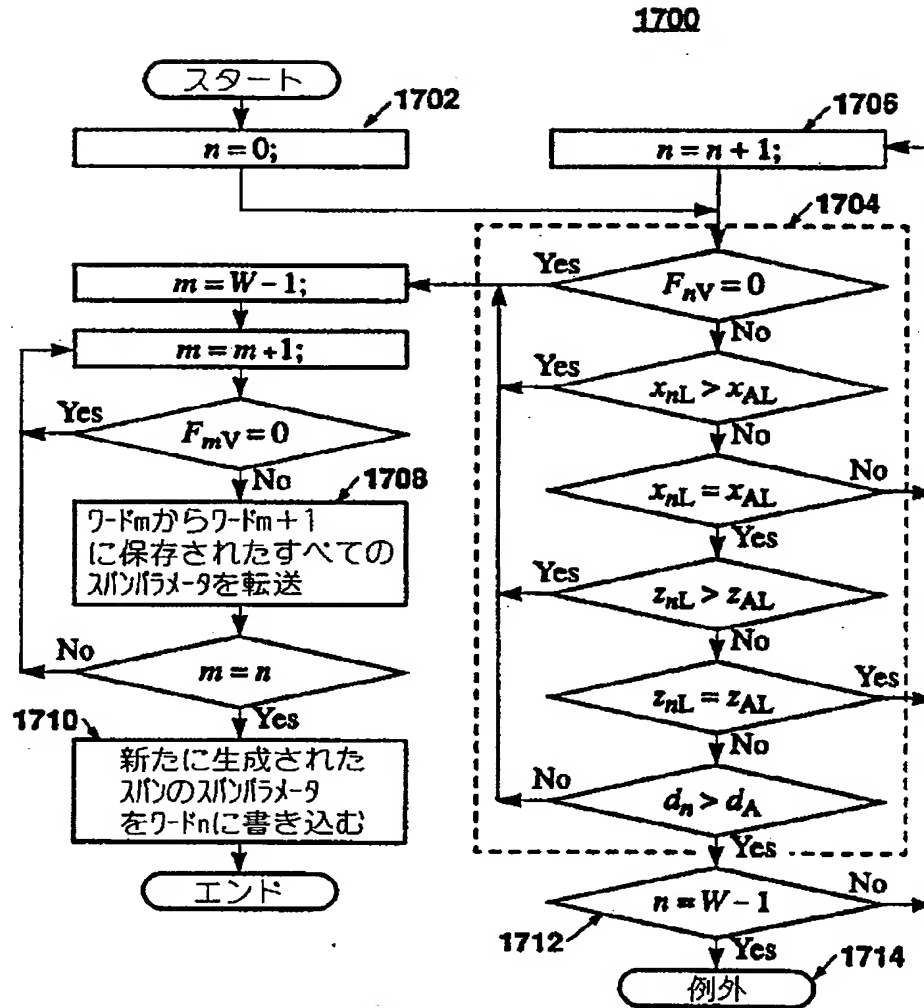
【図16】

Figure 16



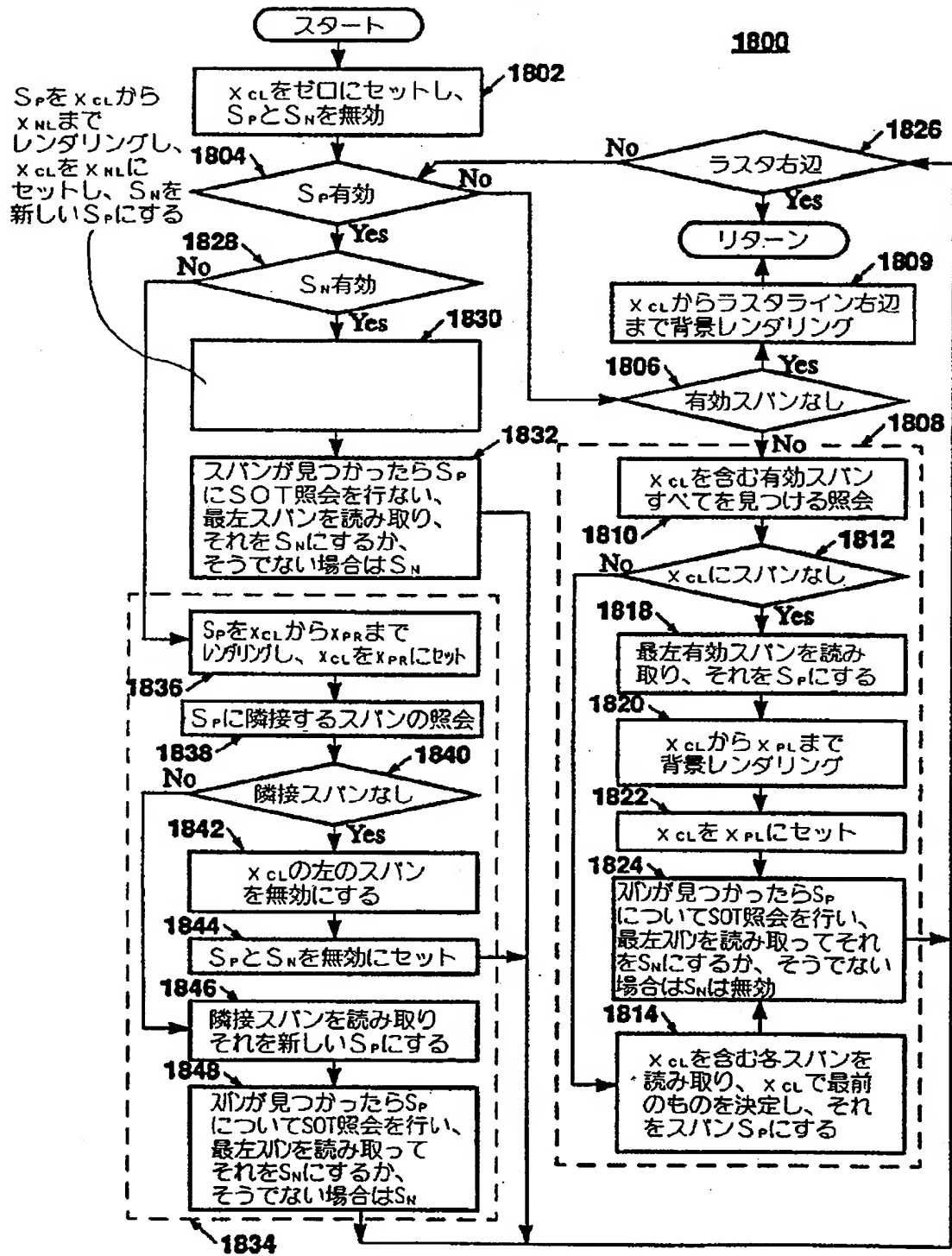
【図 17】

Figure 17

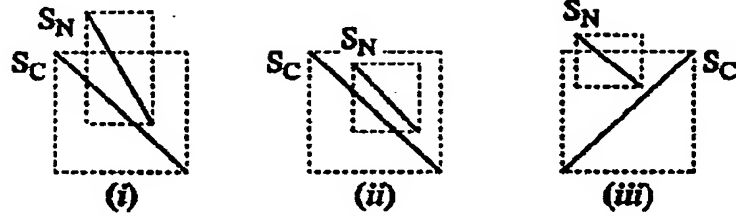
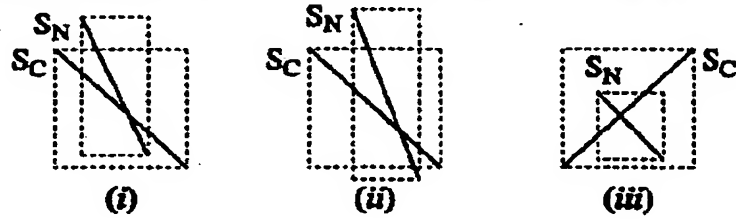
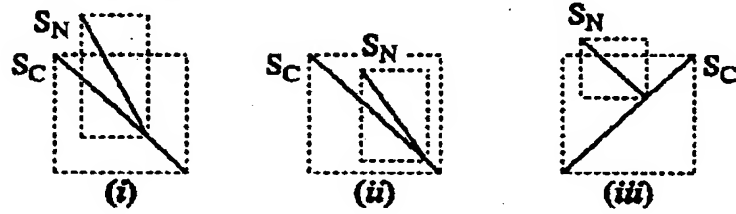
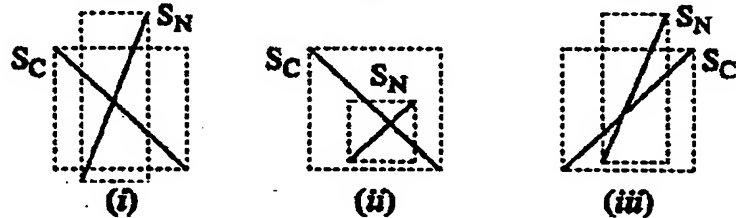
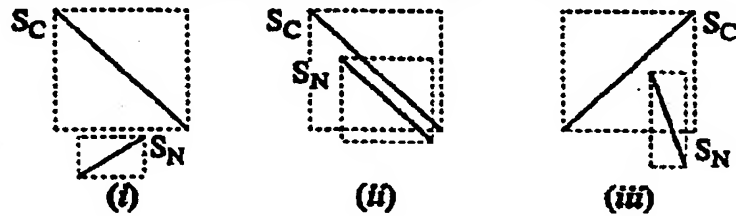


【図18】

Figure 18

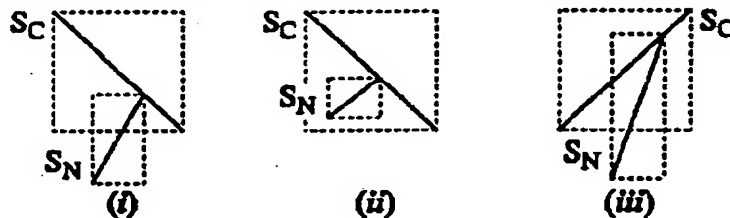
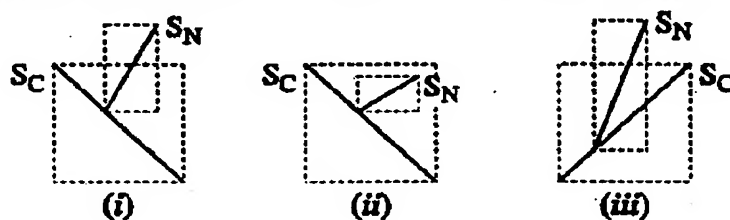
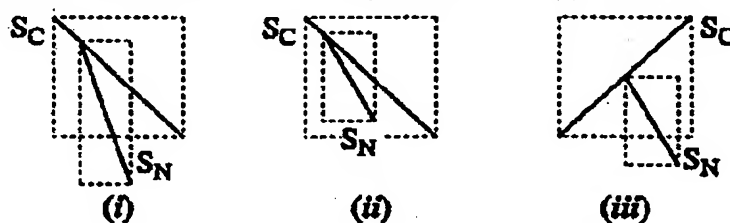
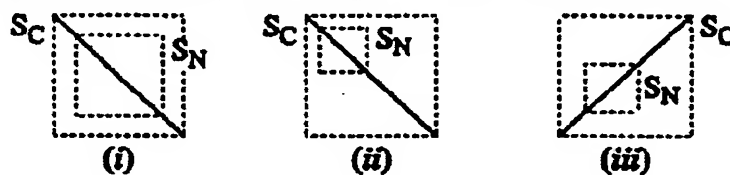
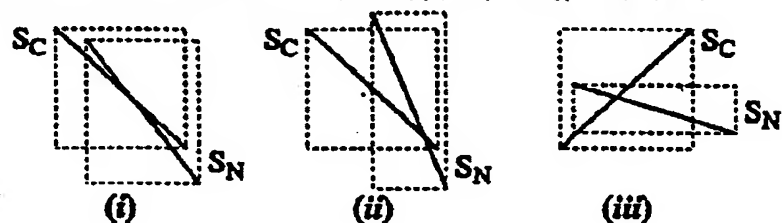


【図 20】

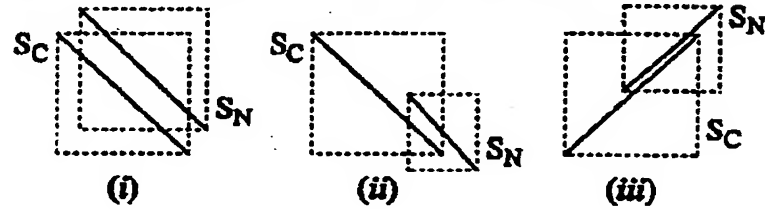
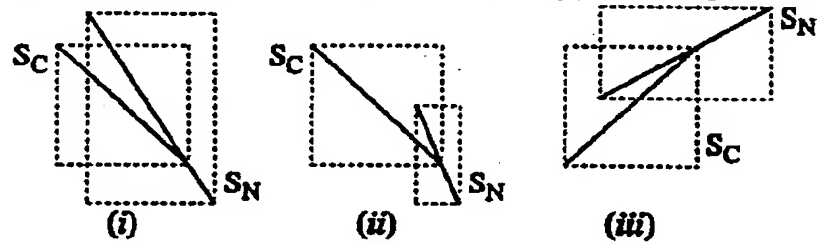
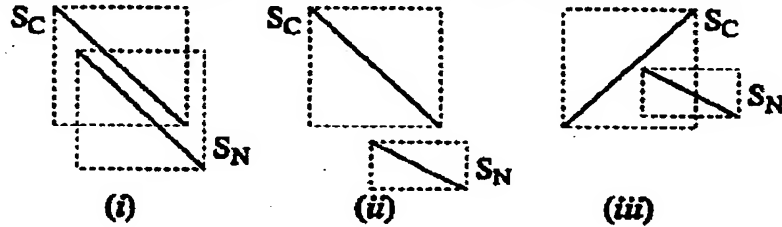
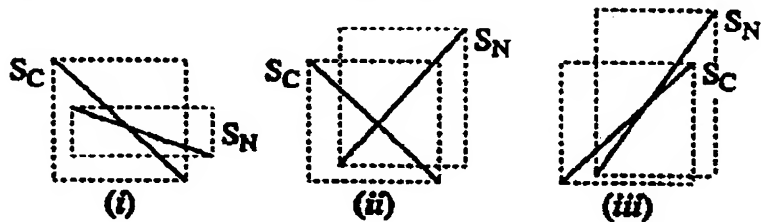
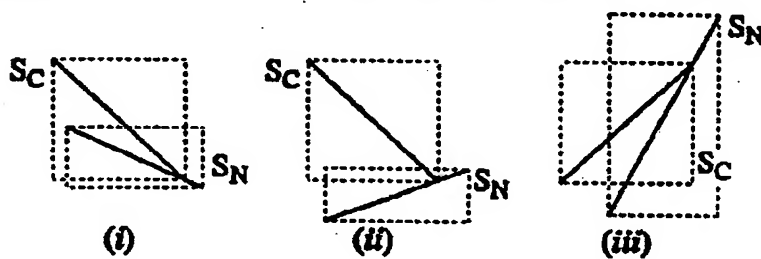
Figure 20**Figure 20A:** インタラクションタイプ 1 = (近, 近, 非, 非) \Rightarrow 1-1 5**Figure 20B:** インタラクションタイプ 2 = (近, 遠, 非, 非) \Rightarrow 1-1 4**Figure 20C:** インタラクションタイプ 3 = (近, 等, 非, 非) \Rightarrow 1-1 5**Figure 20D:** インタラクションタイプ 4 = (遠, 近, 非, 非) \Rightarrow 1-1 3**Figure 20E:** インタラクションタイプ 5 = (遠, 遠, 非, 非) \Rightarrow 1-1 3

【図 20】

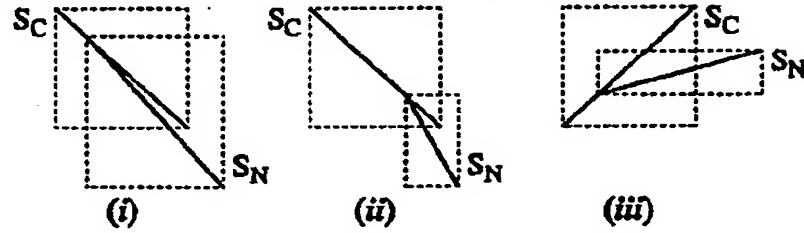
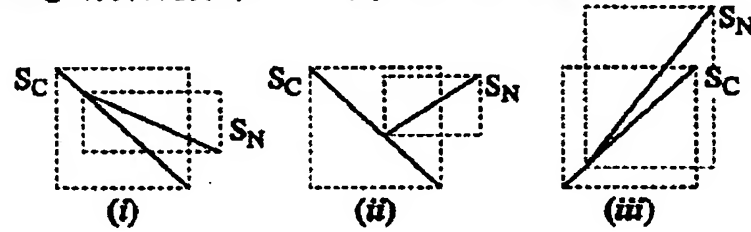
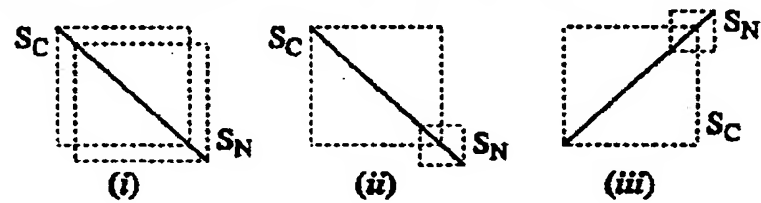
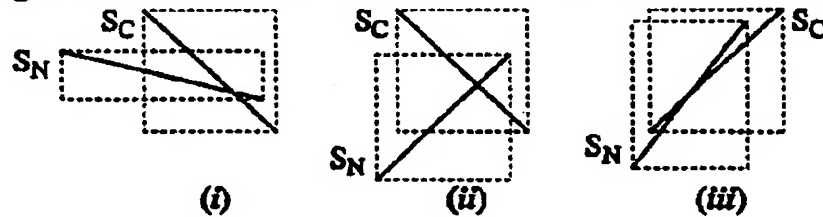
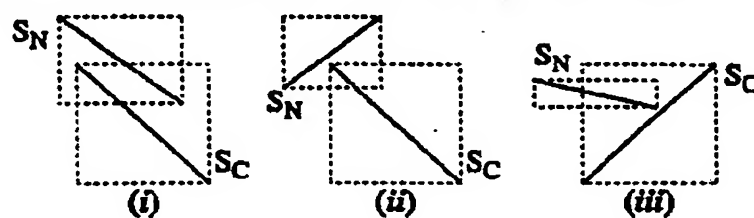
Figure 20 (continued)

Figure 20F: インタクションタイプ 6 = (遠, 等, 非, 非) \Rightarrow 1-1 3Figure 20G: インタクションタイプ 7 = (等, 近, 非, 非) \Rightarrow 1-1 5Figure 20H: インタクションタイプ 8 = (等, 遠, 非, 非) \Rightarrow 1-1 3Figure 20I: インタクションタイプ 9 = (等, 等, 非, 非) \Rightarrow 1-1 5Figure 20J: インタクションタイプ 10 = (近, 非, 非, 近) \Rightarrow 1-1 4

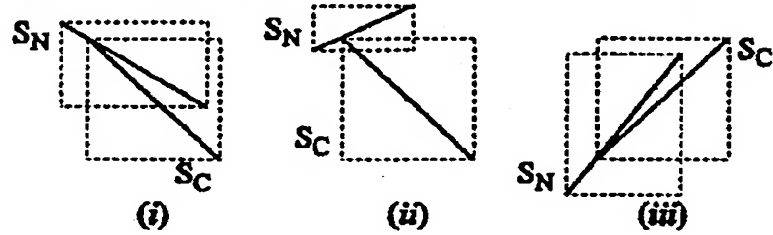
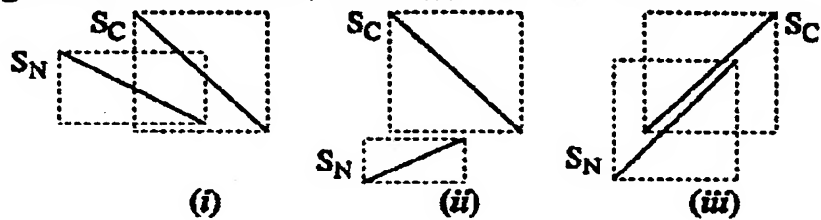
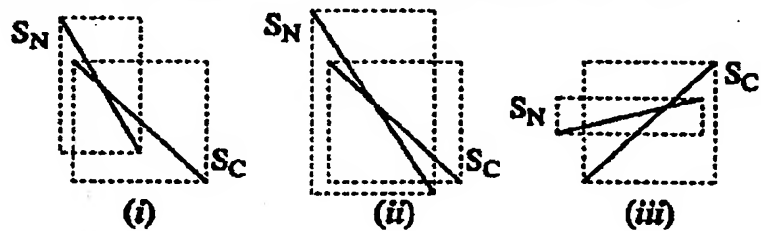
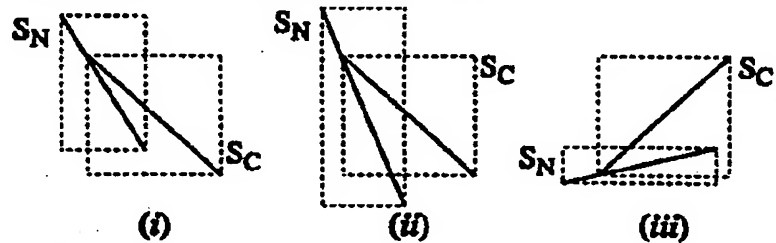
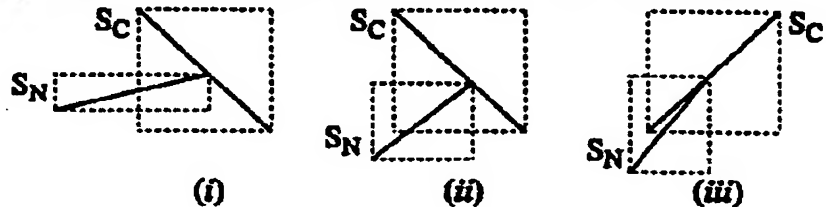
【図 20】

Figure 20 (continued)**Figure 20K:** インタクションタイプ 11 = (近, 非, 非, 遠) \Rightarrow ル-ル 5**Figure 20L:** インタクションタイプ 12 = (近, 非, 非, 等) \Rightarrow ル-ル 5**Figure 20M:** インタクションタイプ 13 = (遠, 非, 非, 近) \Rightarrow ル-ル 3**Figure 20N:** インタクションタイプ 14 = (遠, 非, 非, 近) \Rightarrow ル-ル 3**Figure 20O:** インタクションタイプ 15 = (遠, 非, 非, 等) \Rightarrow ル-ル 3

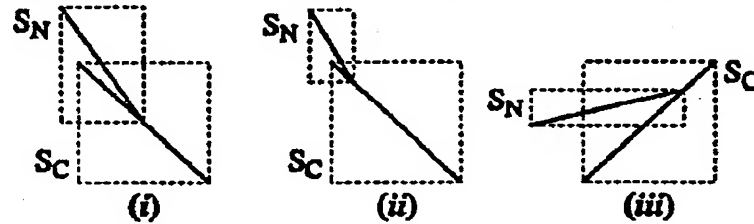
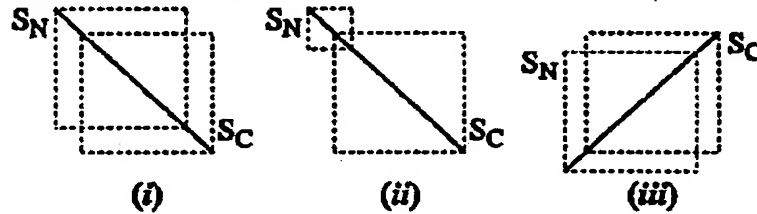
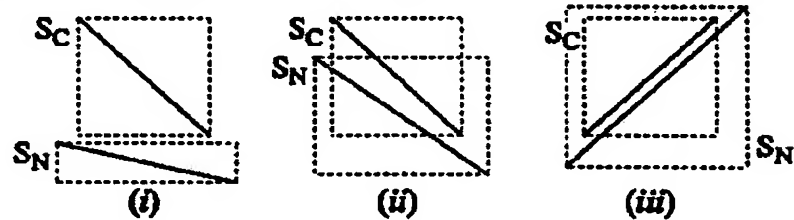
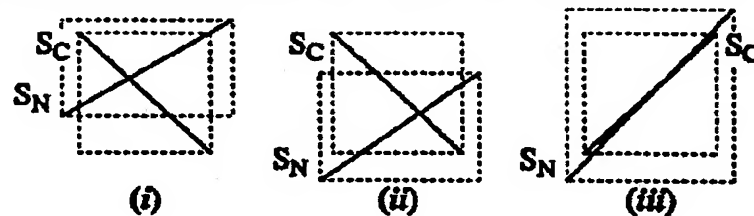
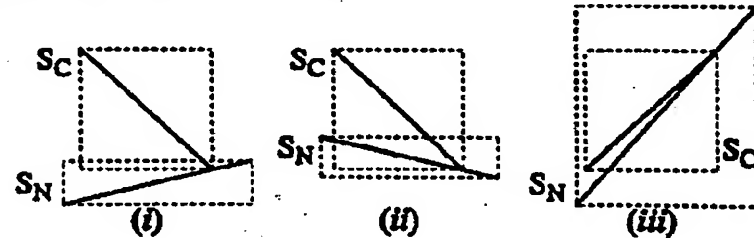
【図 20】

Figure 20 (continued)**Figure 20P:** インタラクションタイプ 16 = (等, 非, 非, 近) \Rightarrow ル-ル 3**Figure 20Q:** インタラクションタイプ 17 = (等, 非, 非, 遠) \Rightarrow ル-ル 5**Figure 20R:** インタラクションタイプ 18 = (等, 非, 非, 等) \Rightarrow ル-ル 5**Figure 20S:** インタラクションタイプ 19 = (非, 近, 近, 非) \Rightarrow 不可**Figure 20T:** インタラクションタイプ 20 = (非, 近, 遠, 非) \Rightarrow ル-ル 5

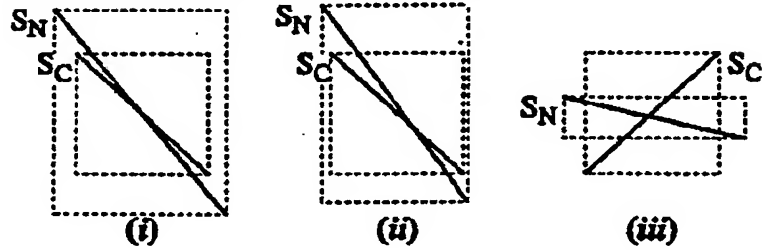
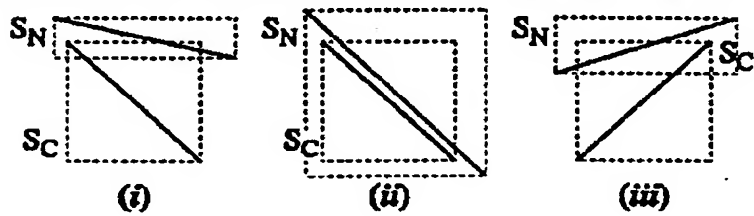
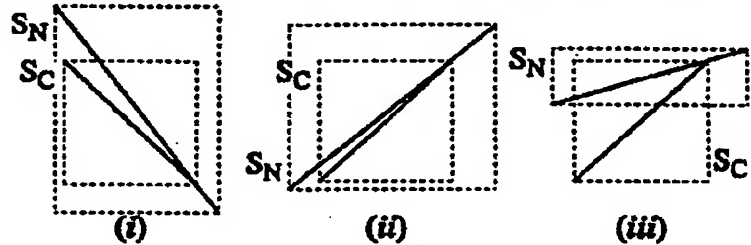
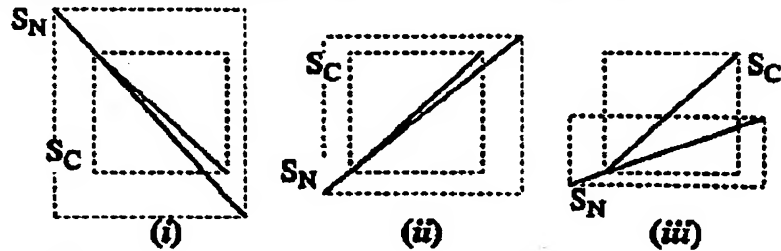
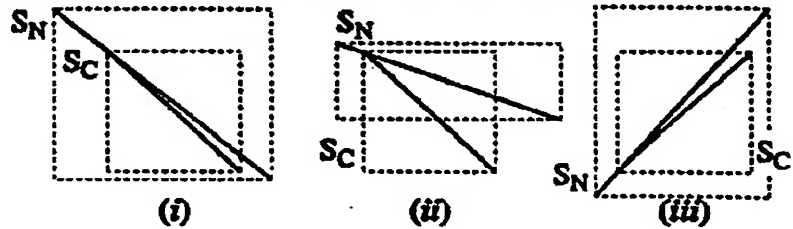
【図 20】

Figure 20 (continued)**Figure 20U:** インタクションタイプ 21 = (非, 近, 等, 非) \Rightarrow ルール 5**Figure 20V:** インタクションタイプ 22 = (非, 遠, 近, 非) \Rightarrow 不可**Figure 20W:** インタクションタイプ 23 = (非, 遠, 遠, 非) \Rightarrow ルール 4**Figure 20X:** インタクションタイプ 24 = (非, 遠, 等, 非) \Rightarrow 不可**Figure 20Y:** インタクションタイプ 25 = (非, 等, 近, 非) \Rightarrow 不可

【図 20】

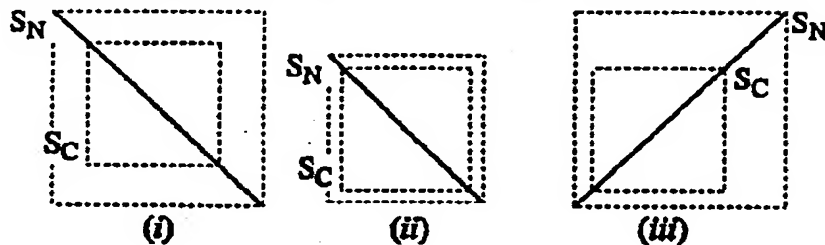
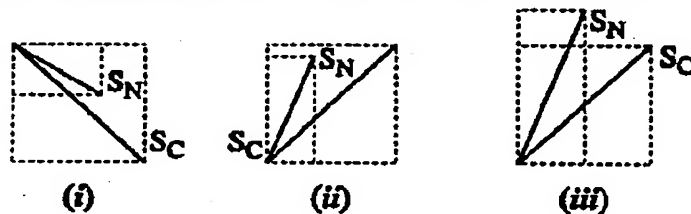
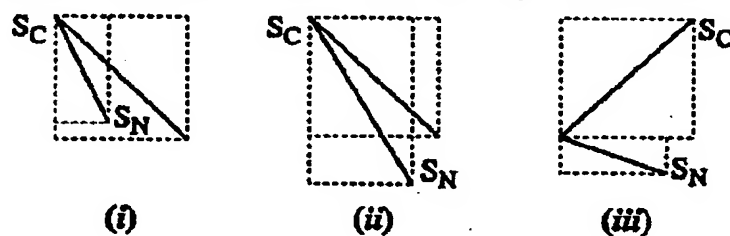
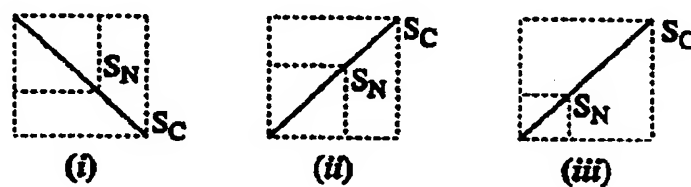
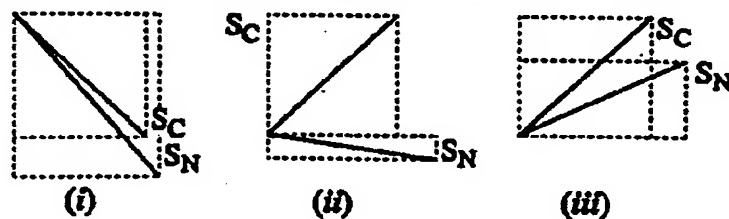
Figure 20 (continued)**Figure 20Z:** インタラクションタイプ 26 = (非, 等, 遠, 非) \Rightarrow ルール 5**Figure 20AA:** インタラクションタイプ 27 = (非, 等, 等, 非) \Rightarrow ルール 5**Figure 20BB:** インタラクションタイプ 28 = (非, 非, 近, 近) \Rightarrow 不可**Figure 20CC:** インタラクションタイプ 29 = (非, 非, 近, 遠) \Rightarrow 不可**Figure 20DD:** インタラクションタイプ 30 = (非, 非, 近, 等) \Rightarrow 不可

【図 20】

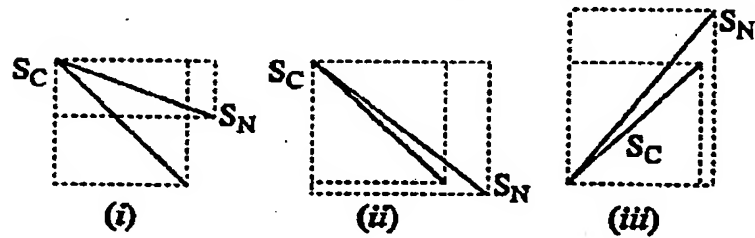
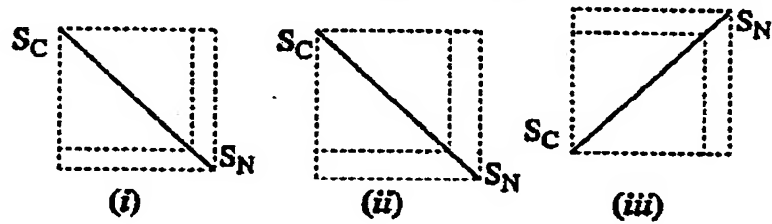
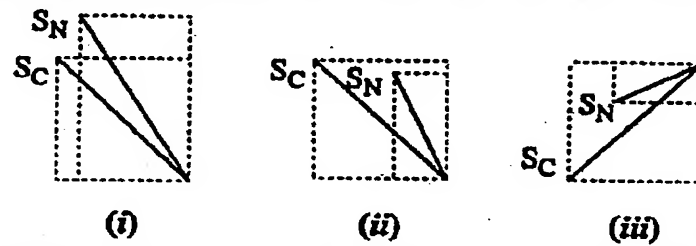
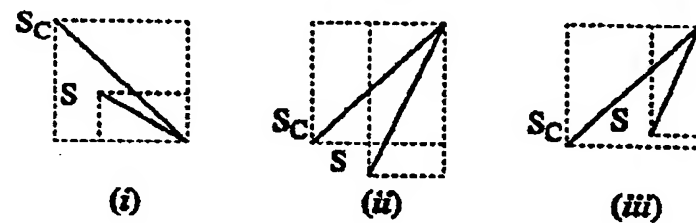
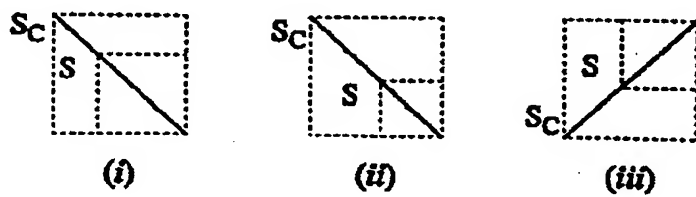
Figure 20 (continued)**Figure 20EE:** インタラクションタイプ 31 = (非, 非, 遠, 近) \Rightarrow ル-ル 4**Figure 20FF:** インタラクションタイプ 32 = (非, 非, 遠, 遠) \Rightarrow ル-ル 5**Figure 20GG:** インタラクションタイプ 33 = (非, 非, 遠, 等) \Rightarrow ル-ル 5**Figure 20HH:** インタラクションタイプ 34 = (非, 非, 等, 近) \Rightarrow 不可**Figure 20II:** インタラクションタイプ 35 = (非, 非, 等, 遠) \Rightarrow ル-ル 5

【図 20】

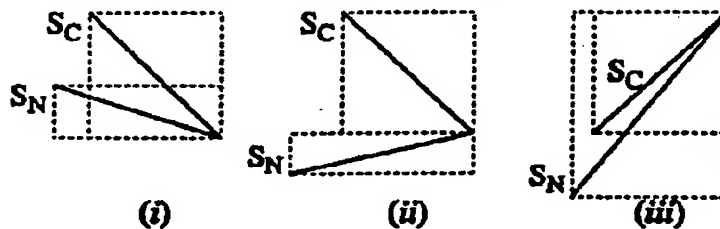
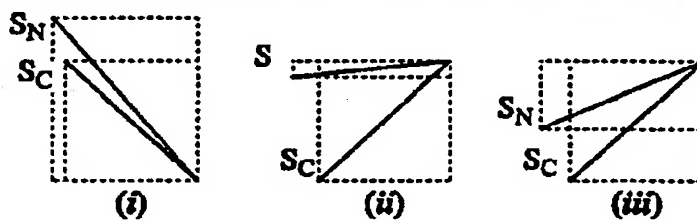
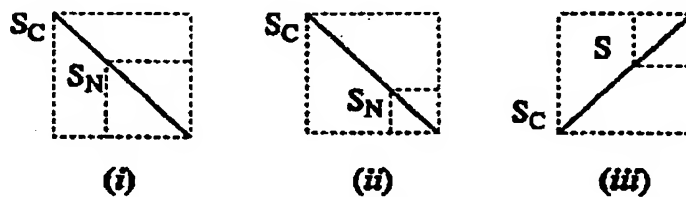
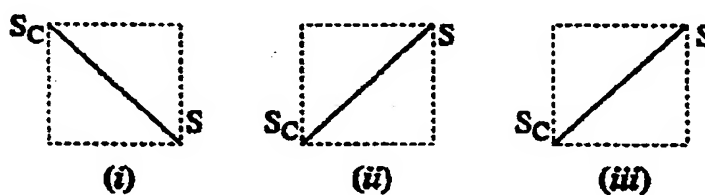
Figure 20 (continued)

Figure 20JJ: インタラクションタイプ 36 = (非, 非, 等, 等) \Rightarrow ルール 5Figure 20KK: インタラクションタイプ 37 = (等, 近, 等, 非) \Rightarrow ルール 5Figure 20LL: インタラクションタイプ 38 = (等, 遠, 等, 非) \Rightarrow 不可Figure 20MM: インタラクションタイプ 39 = (等, 等, 等, 非) \Rightarrow ルール 5Figure 20NN: インタラクションタイプ 40 = (等, 非, 等, 近) \Rightarrow 不可

【図 20】

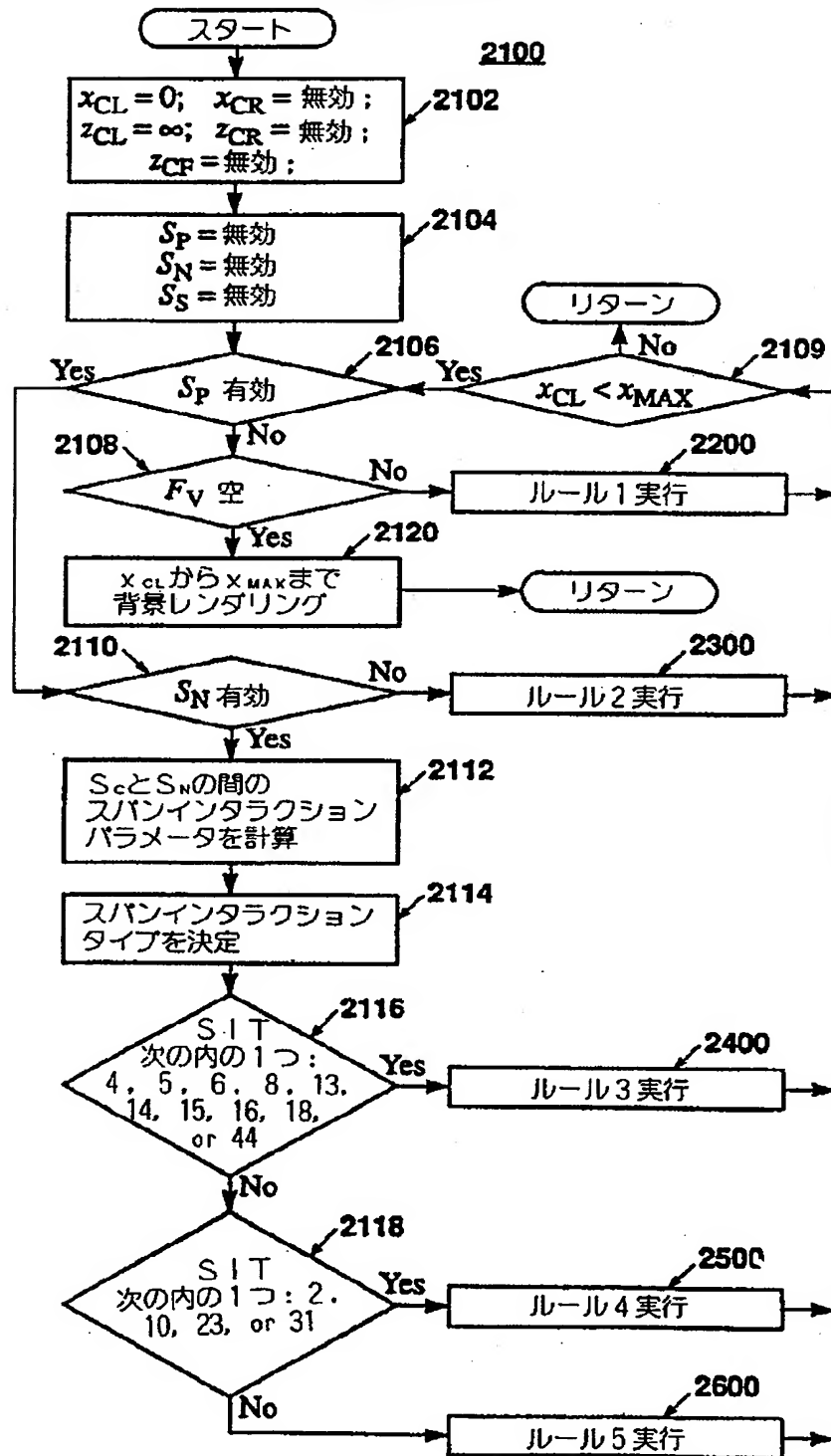
Figure 20 (continued)**Figure 20OO:** インタラクションタイプ 41 = (等, 非, 等, 遠) \Rightarrow 1-1 5**Figure 20PP:** インタラクションタイプ 42 = (等, 非, 等, 等) \Rightarrow 1-1 5**Figure 20QQ:** インタラクションタイプ 43 = (近, 等, 非, 等) \Rightarrow 1-1 5**Figure 20RR:** インタラクションタイプ 44 = (遠, 等, 非, 等) \Rightarrow 1-1 3**Figure 20SS:** インタラクションタイプ 45 = (等, 等, 非, 等) \Rightarrow 1-1 5

【図 20】

Figure 20 (continued)**Figure 20TT:** インタクションタイプ 46 = (非, 等, 近, 等) \Rightarrow 不可**Figure 20UU:** インタクションタイプ 47 = (非, 等, 遠, 等) \Rightarrow 不可 5**Figure 20VV:** インタクションタイプ 48 = (非, 等, 等, 等) \Rightarrow 不可 5**Figure 20WW:** インタクションタイプ 49 = (等, 等, 等, 等) \Rightarrow 不可 5

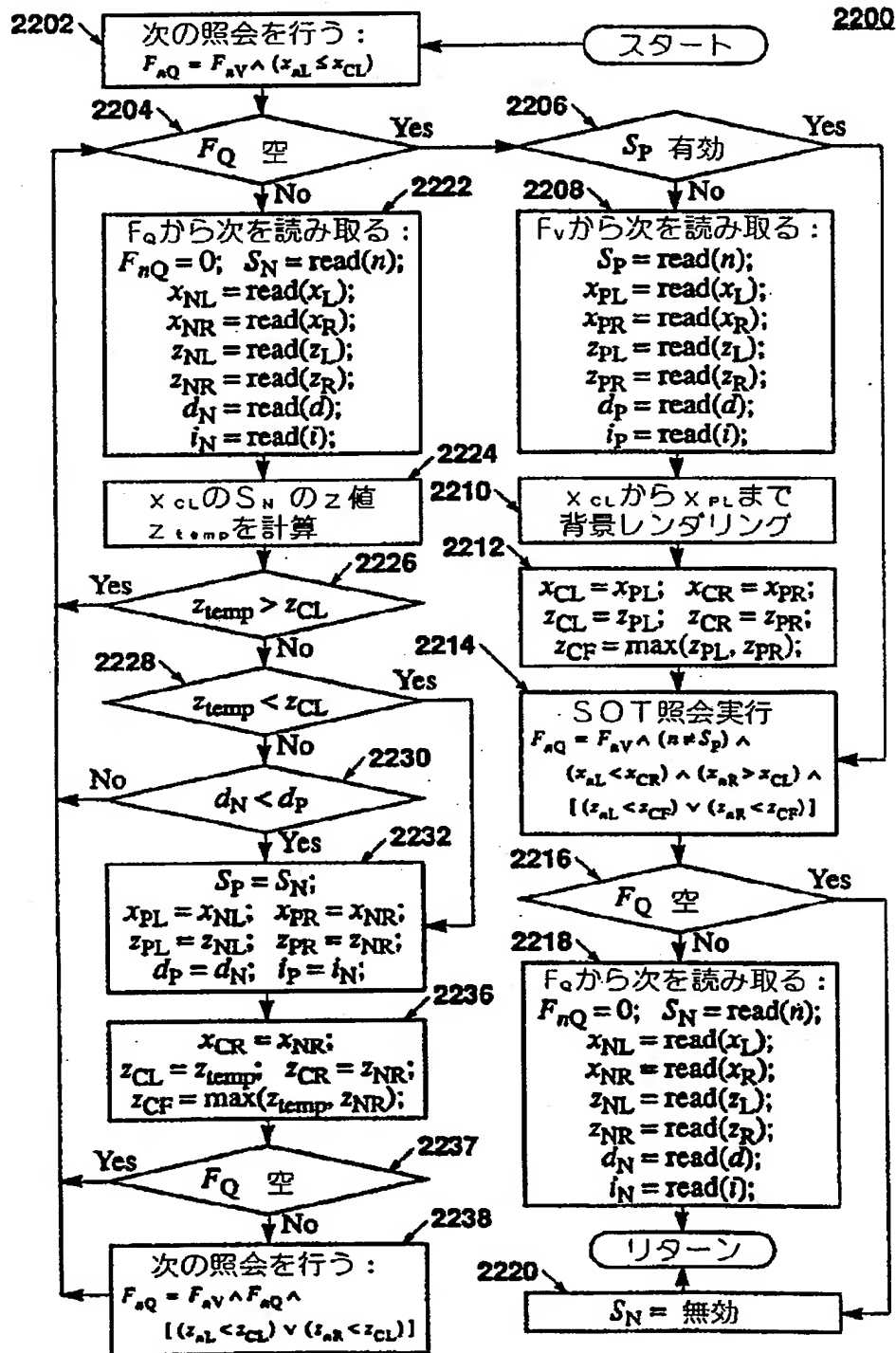
【図 2 1】

Figure 21



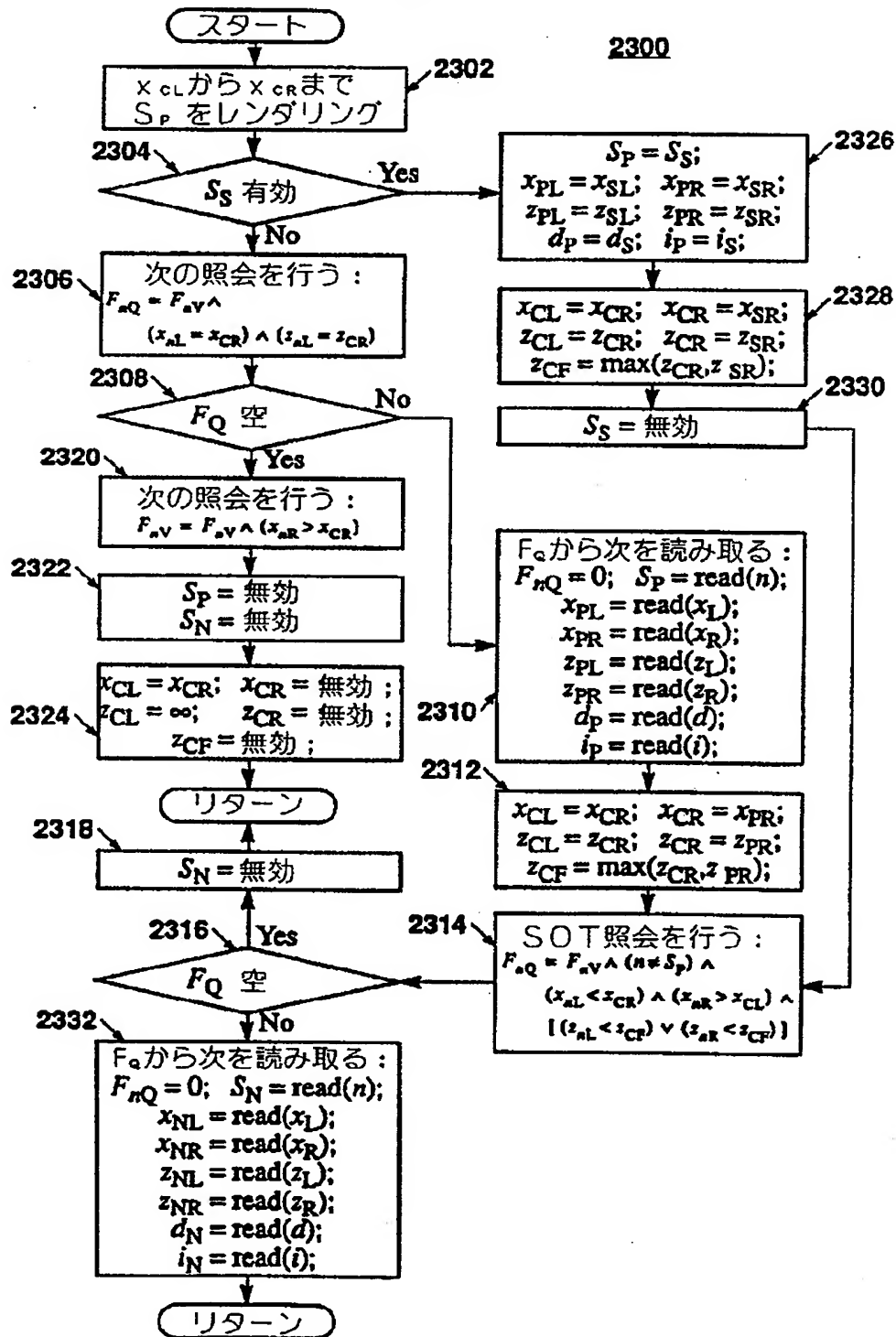
【図 22】

Figure 22



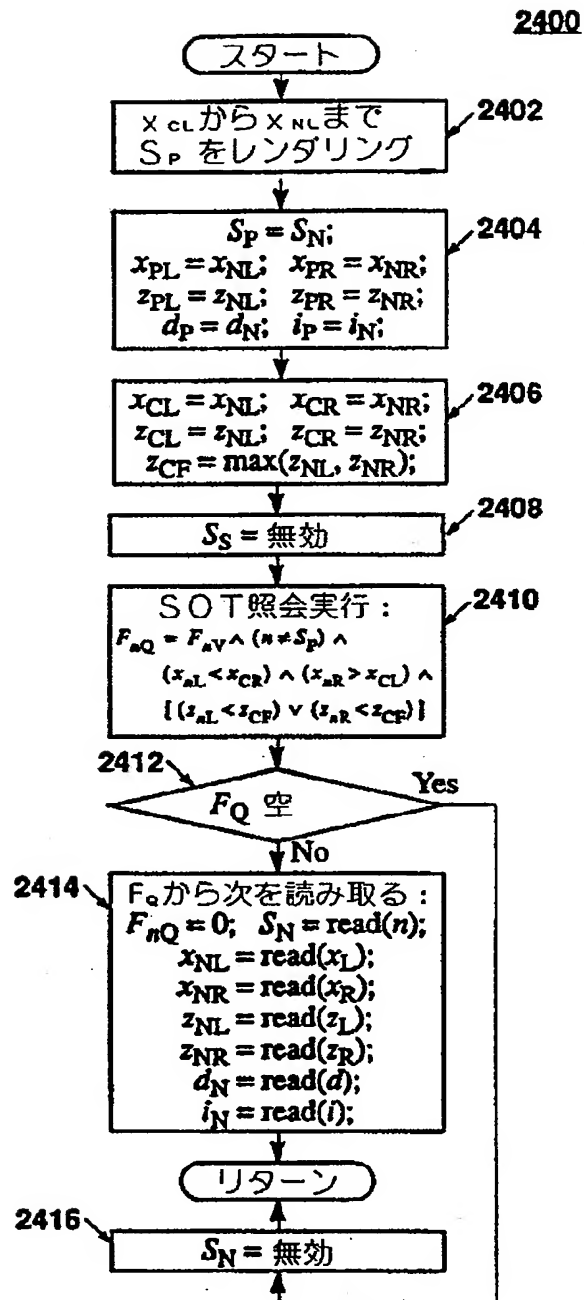
【図 23】

Figure 23



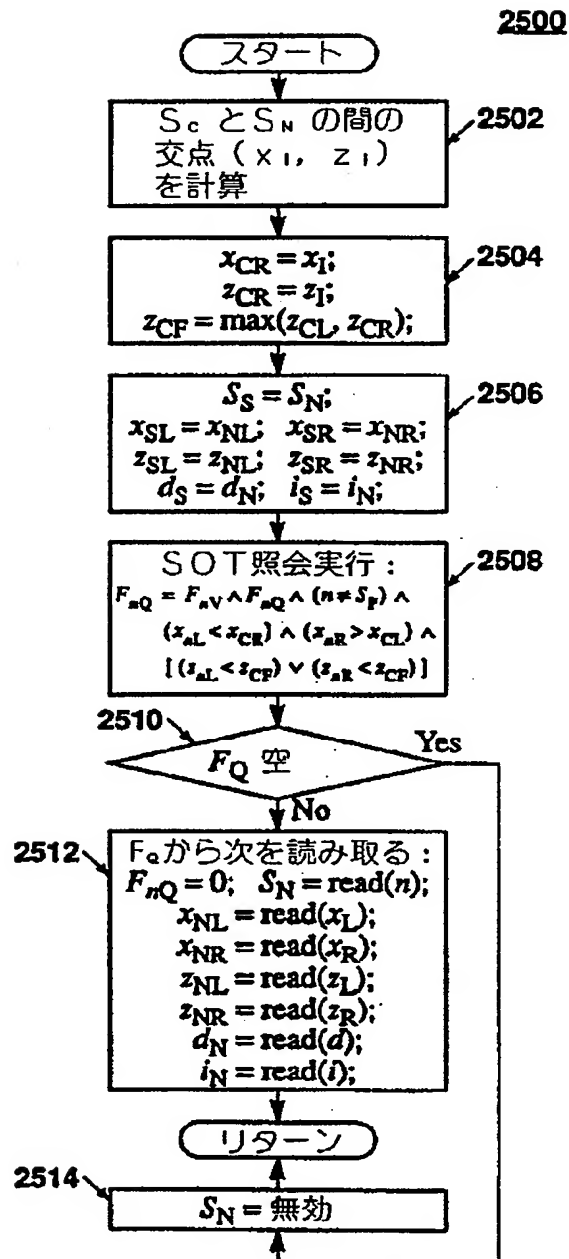
【図 2 4】

Figure 24



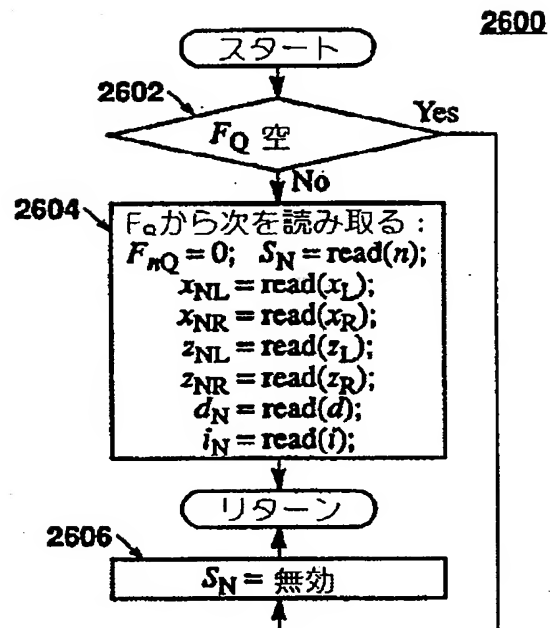
【図25】

Figure 25



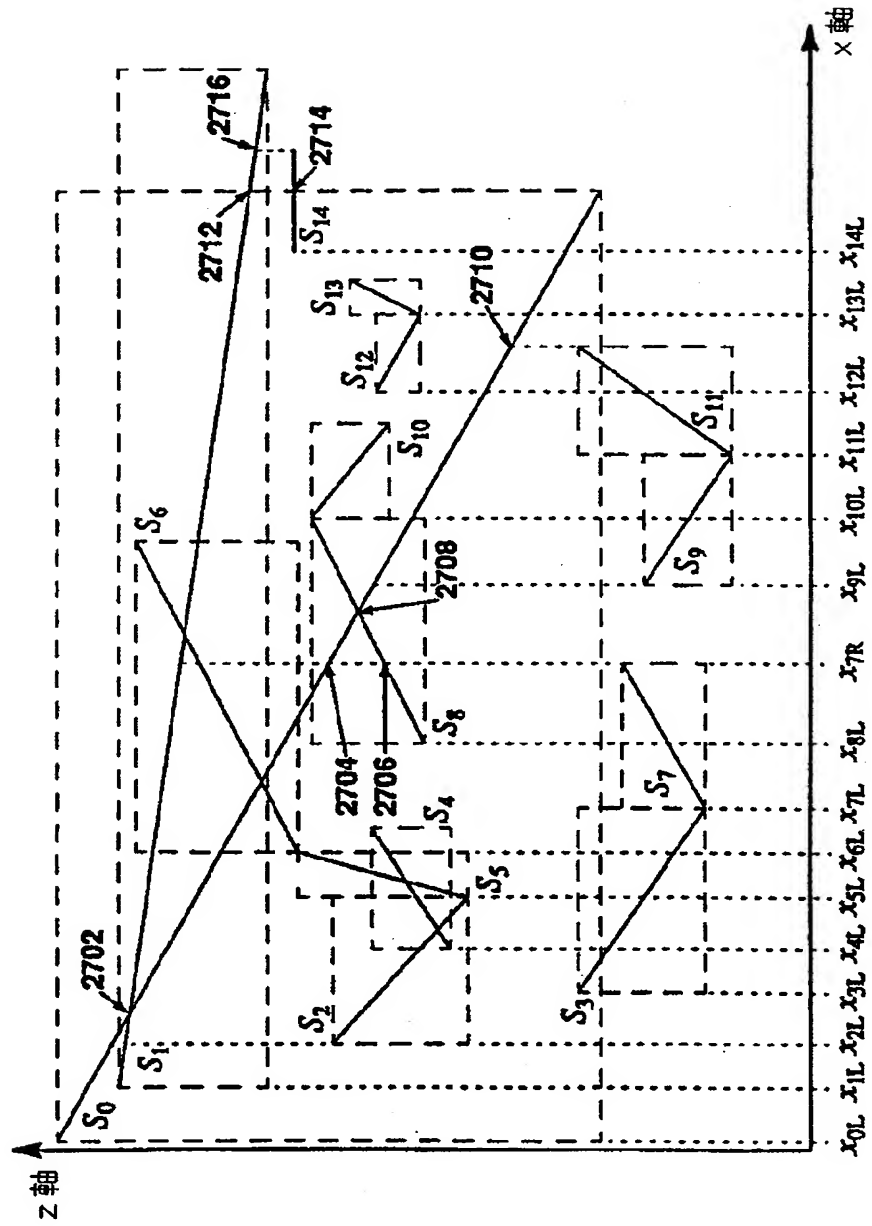
【図26】

Figure 26

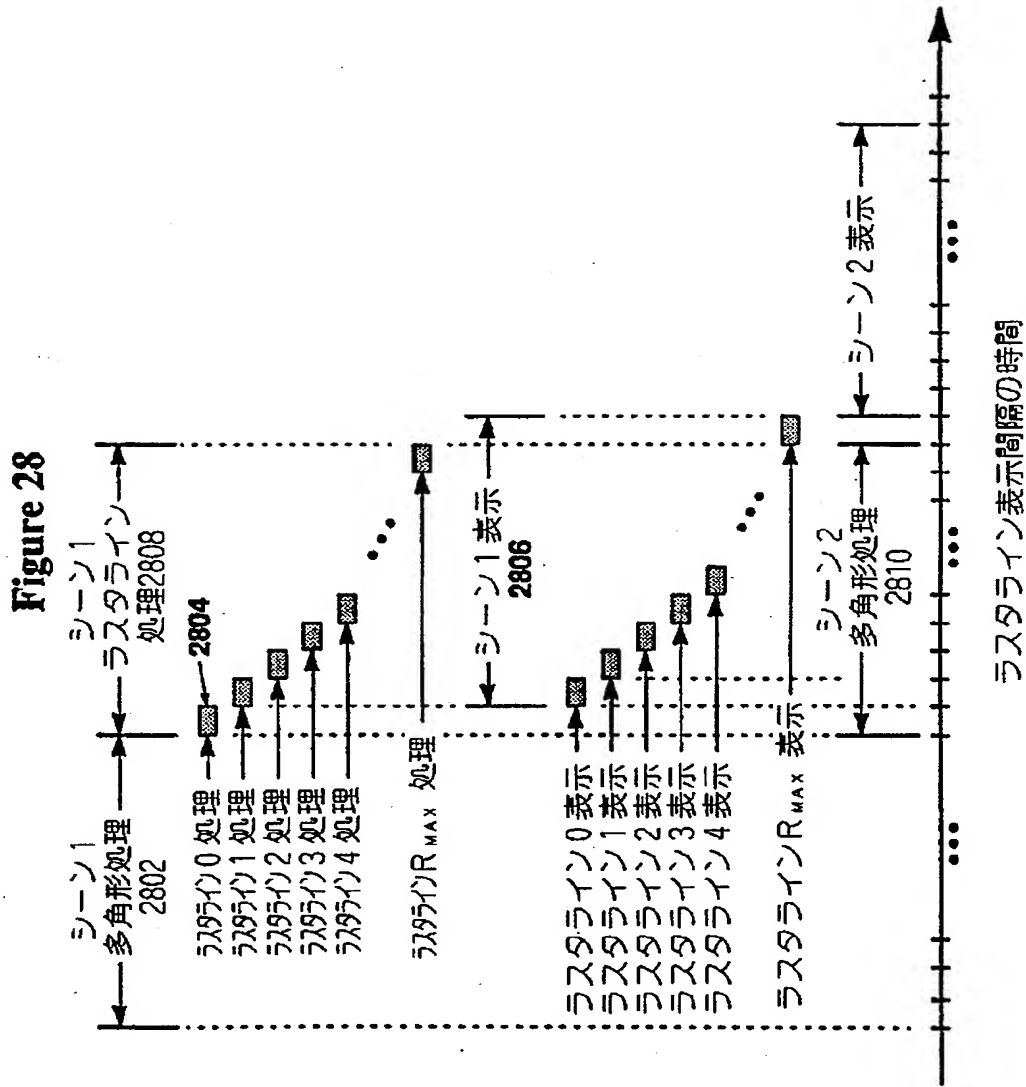


【图 27】

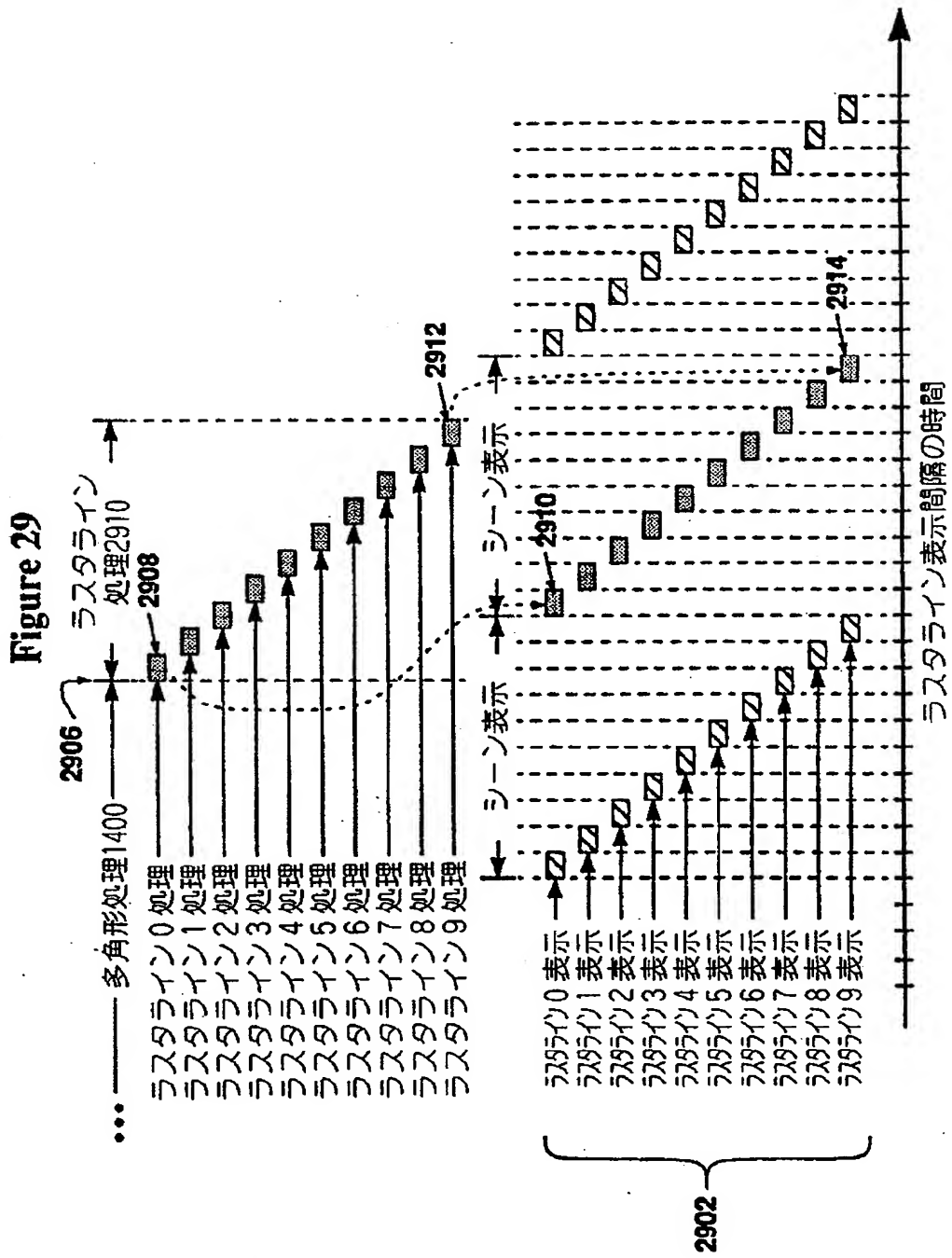
Figure 27



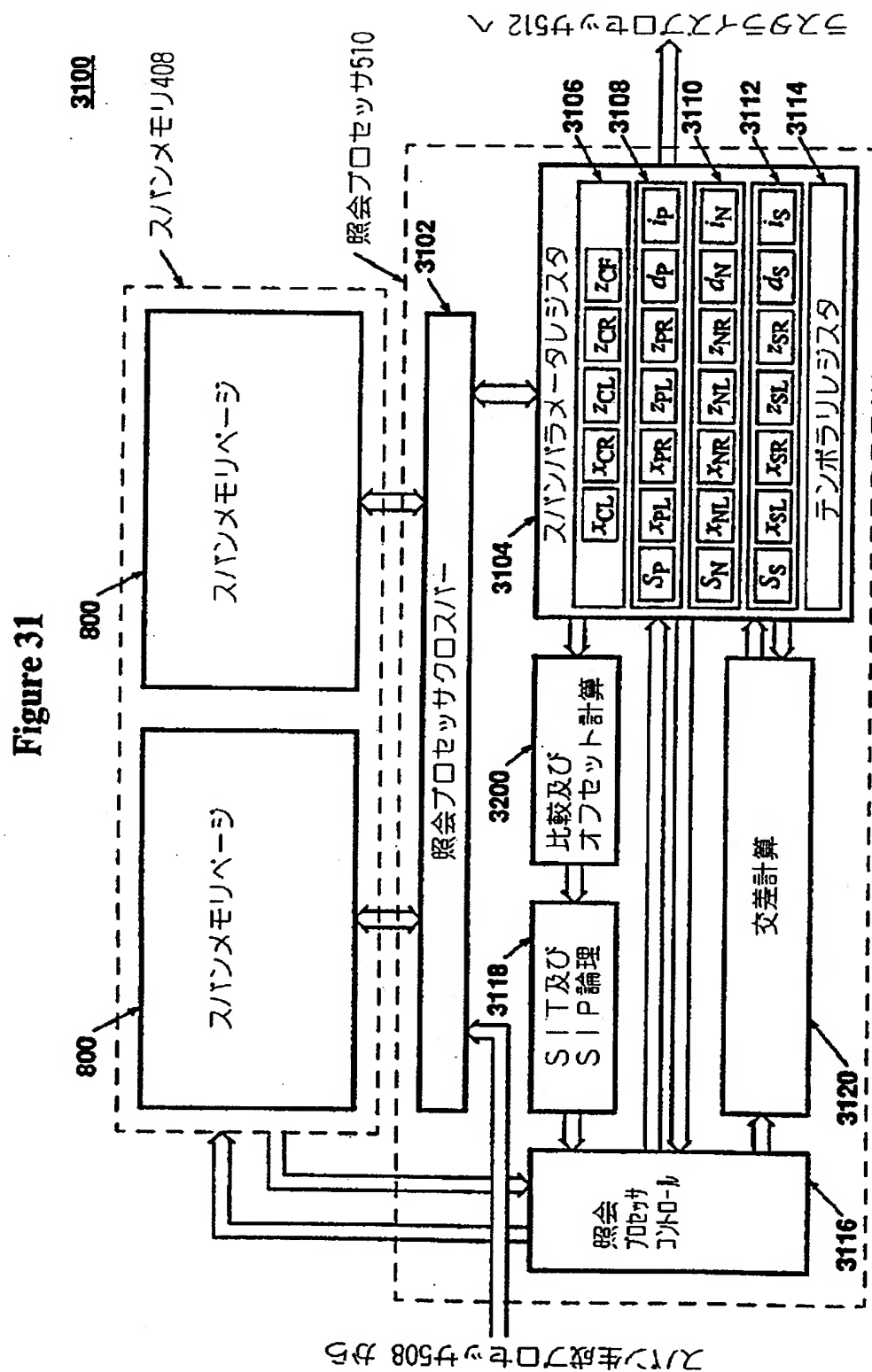
【図 28】



【図 29】



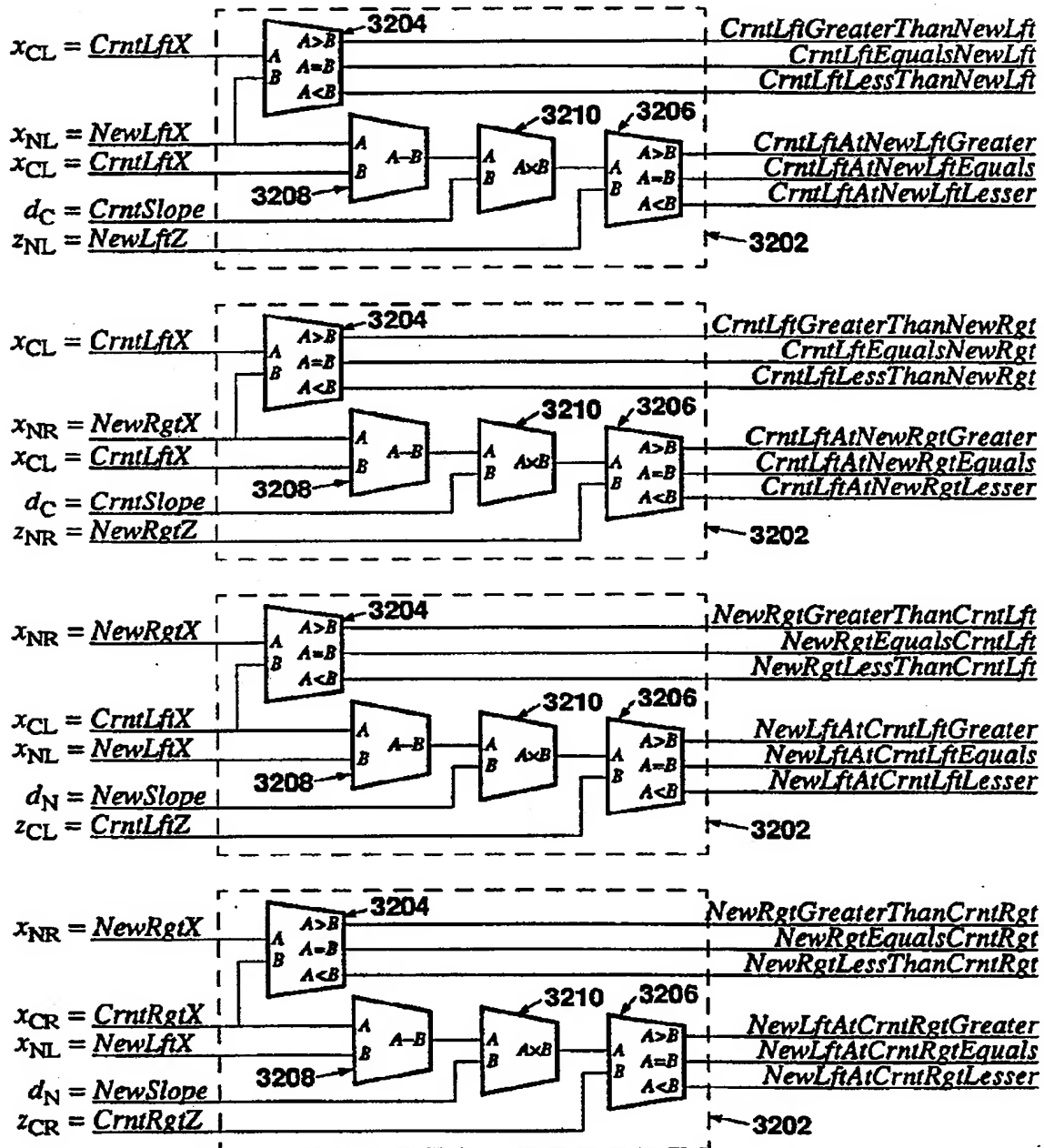
【図 3 1】



【図 3 2】

Figure 32

3200

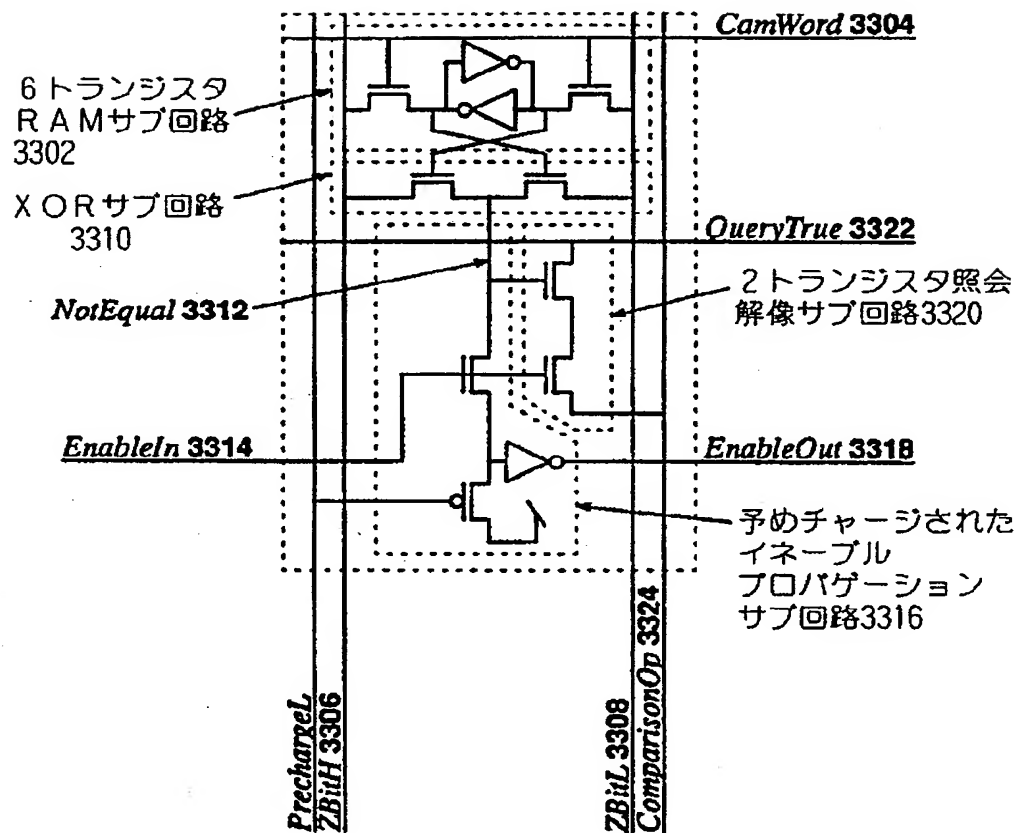


【図33】

Figure 33

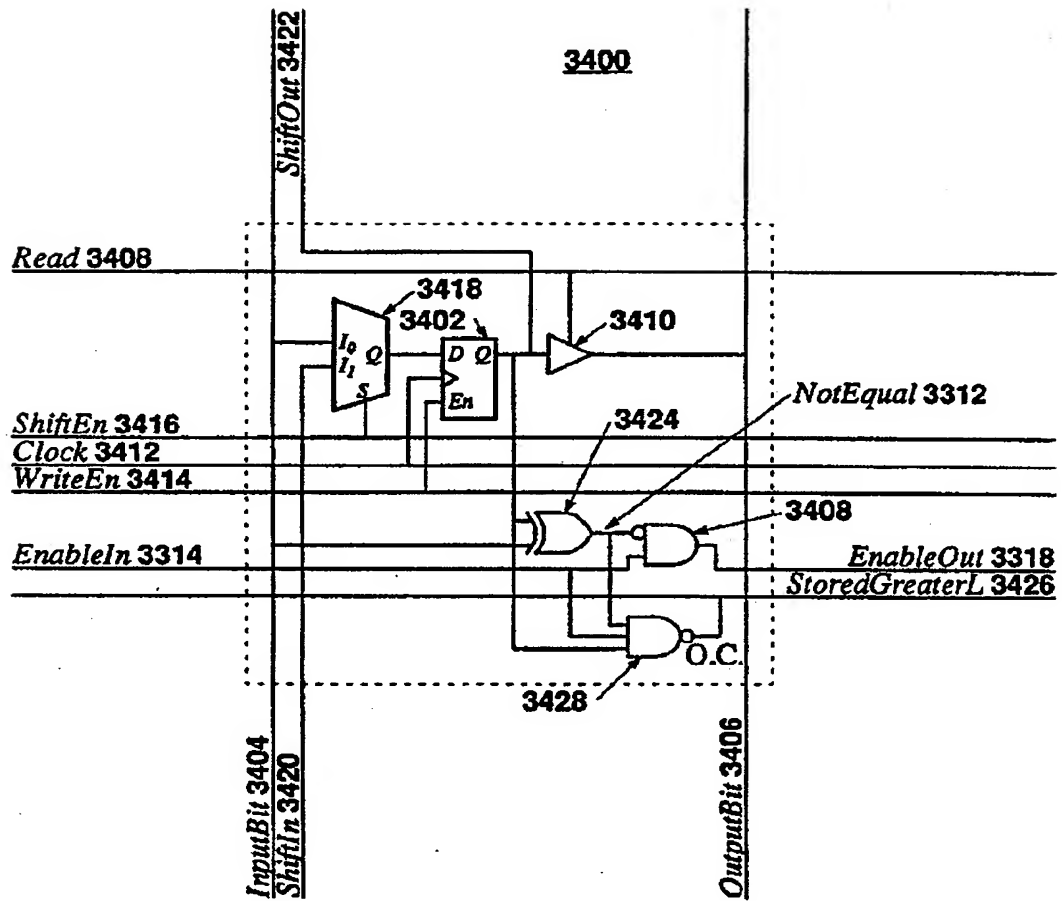
3300

先行技術



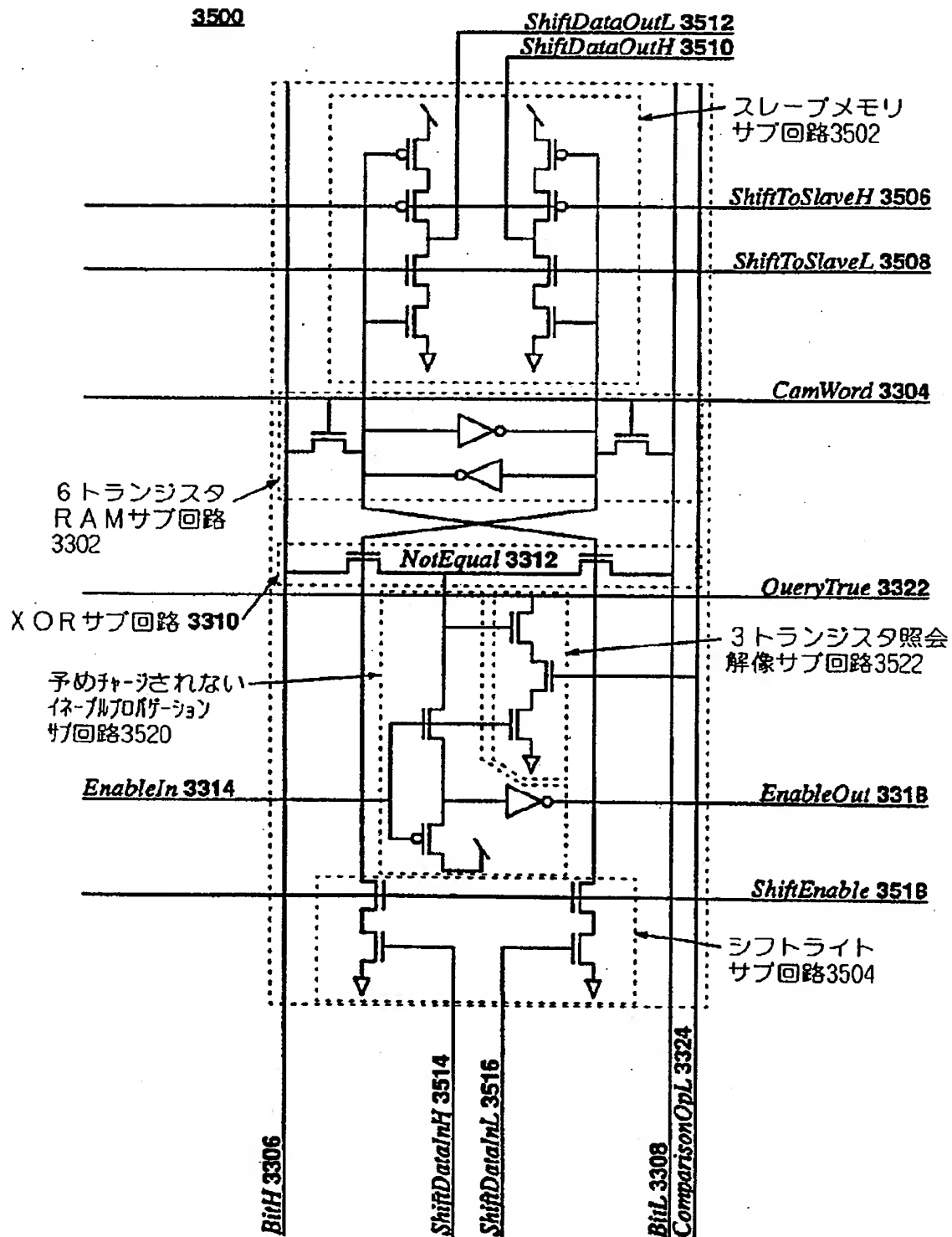
【图 3 4】

Figure 34



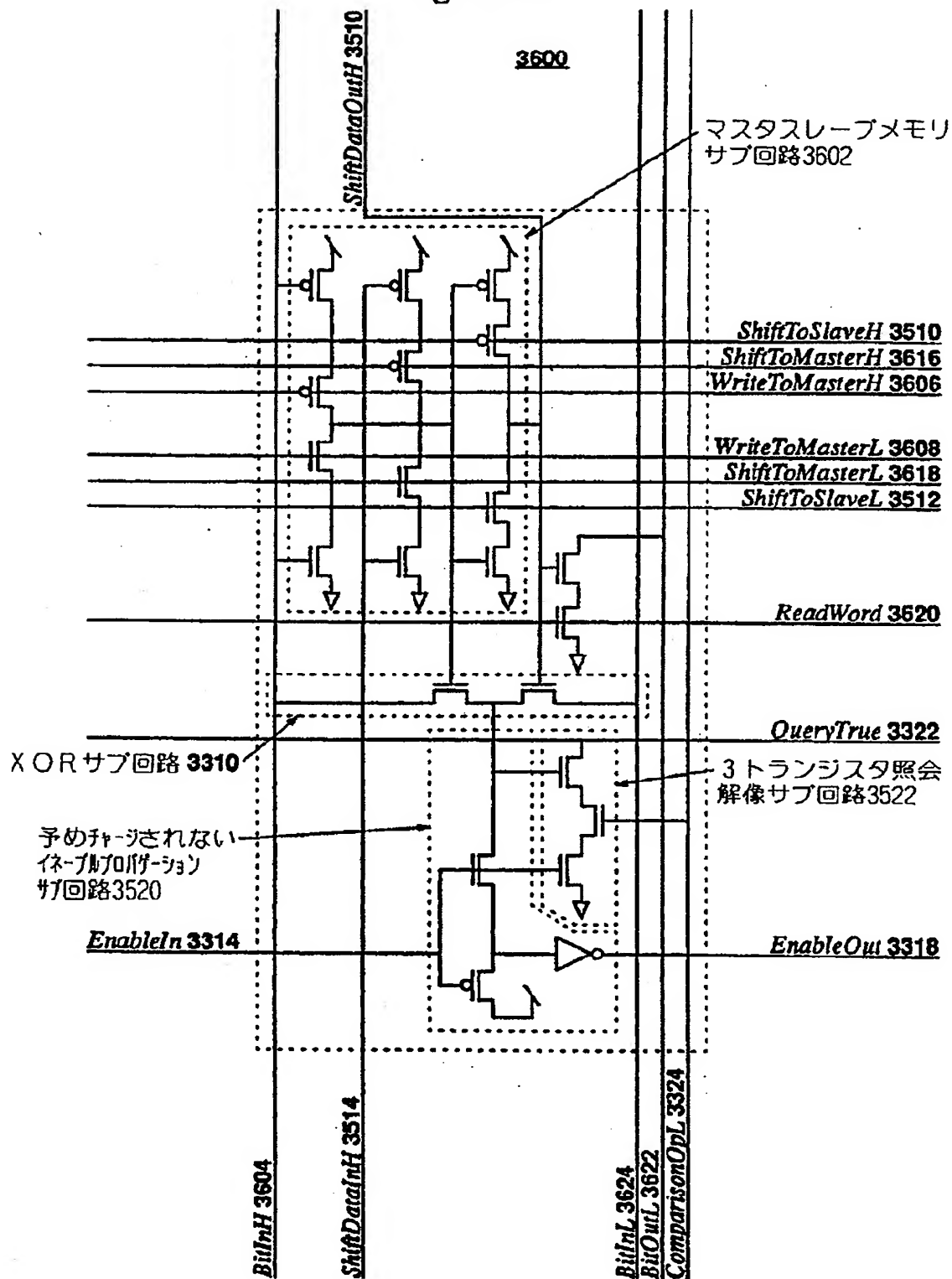
【図35】

Figure 35



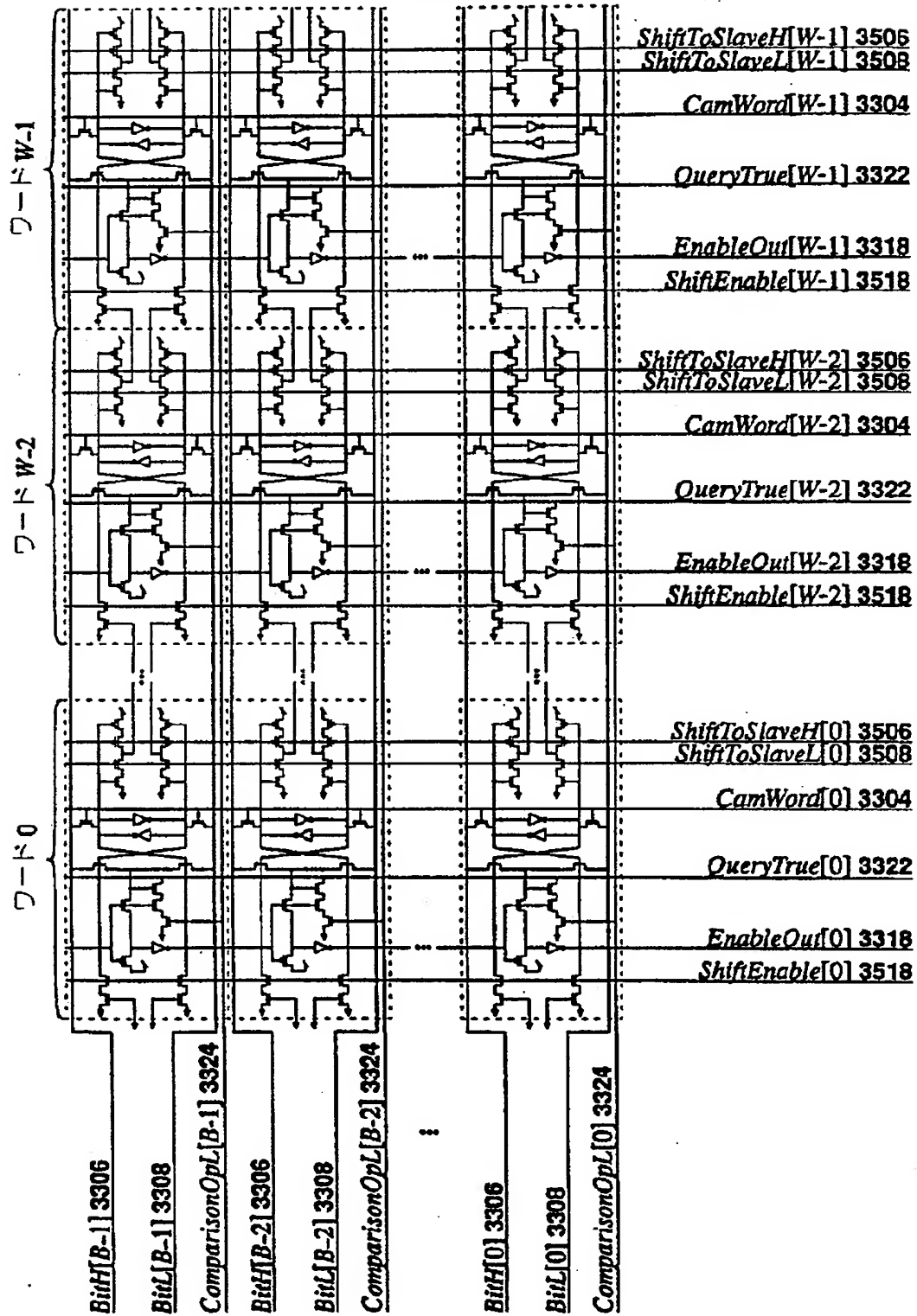
【図 3 6】

Figure 36



【図 37】

Figure 37



【図 3 8】

Figure 38

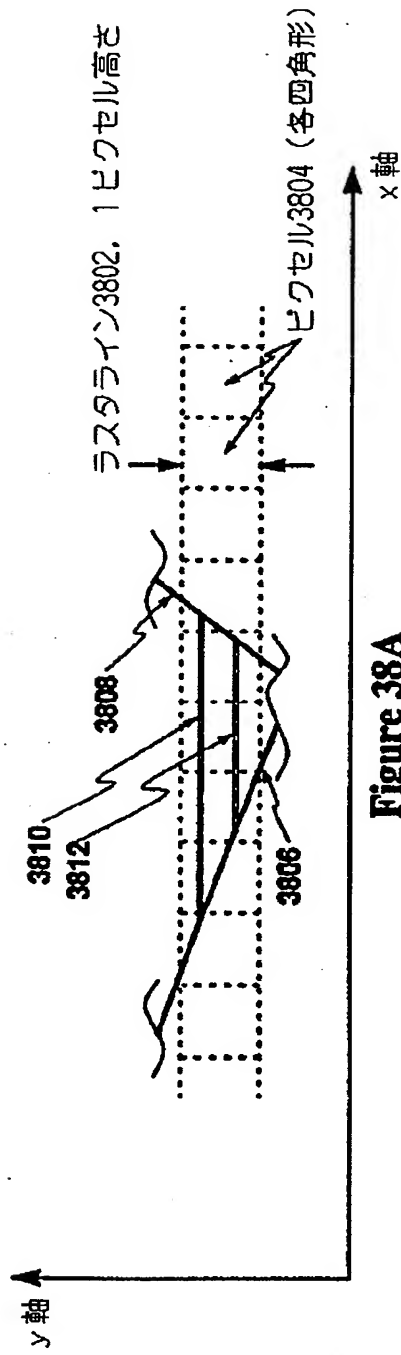


Figure 38A

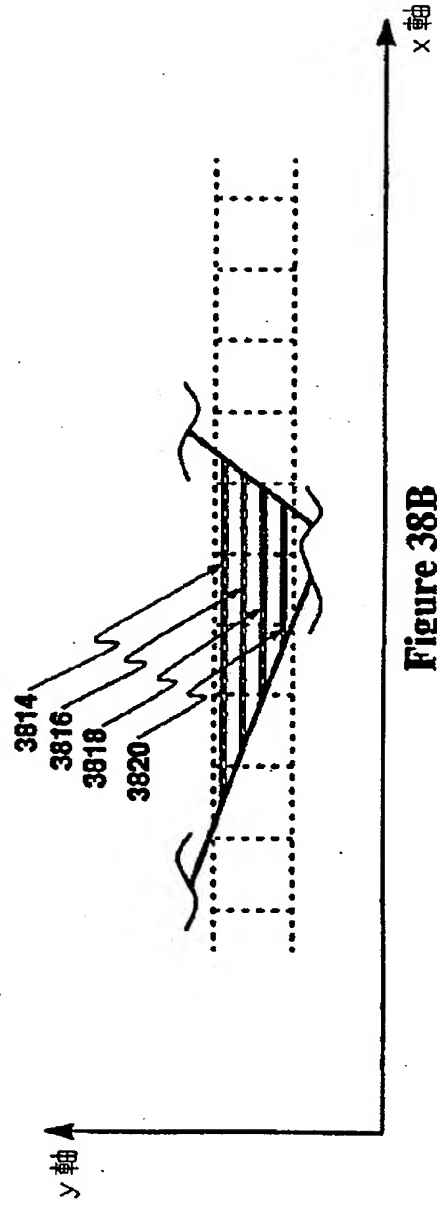
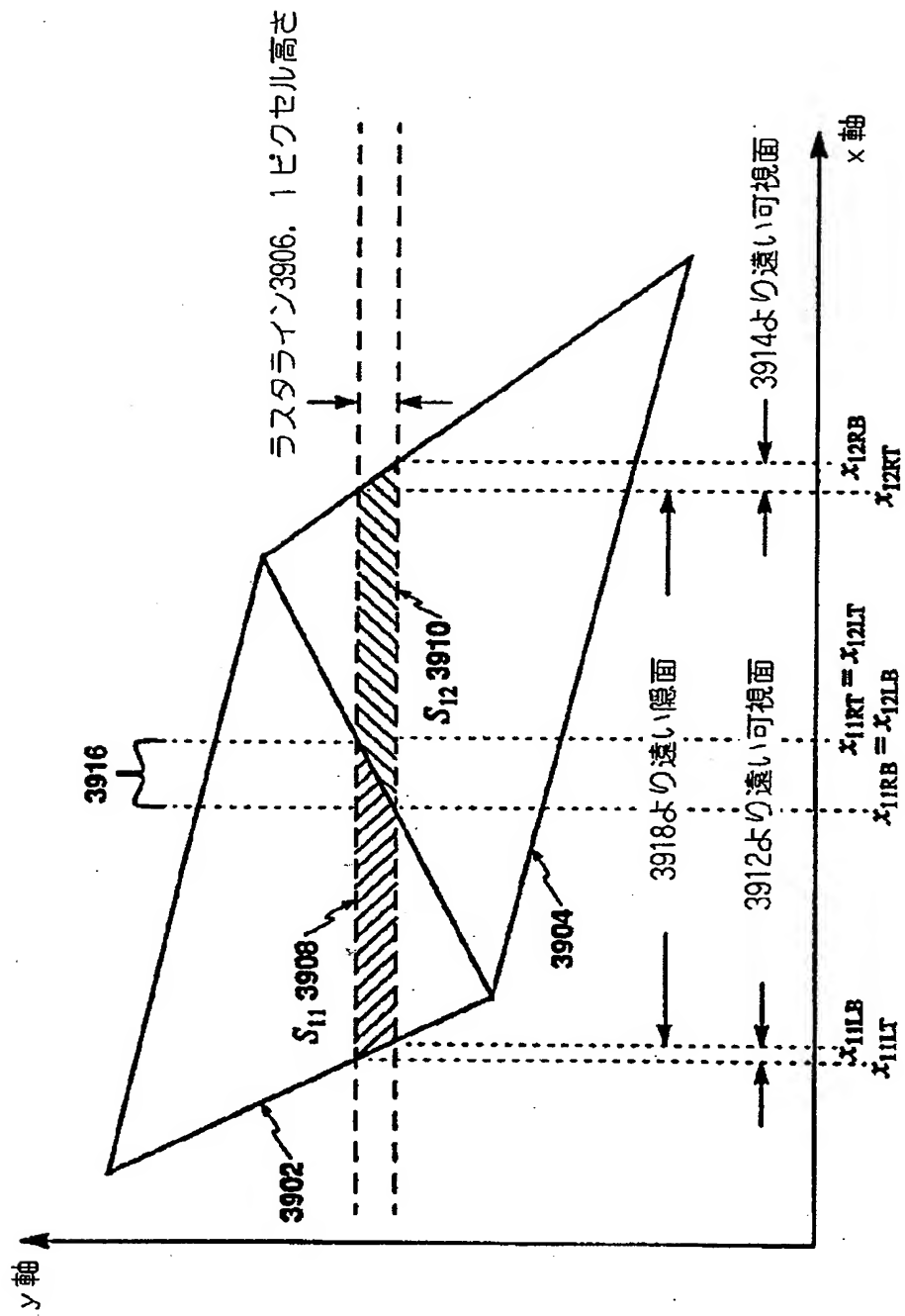


Figure 38B

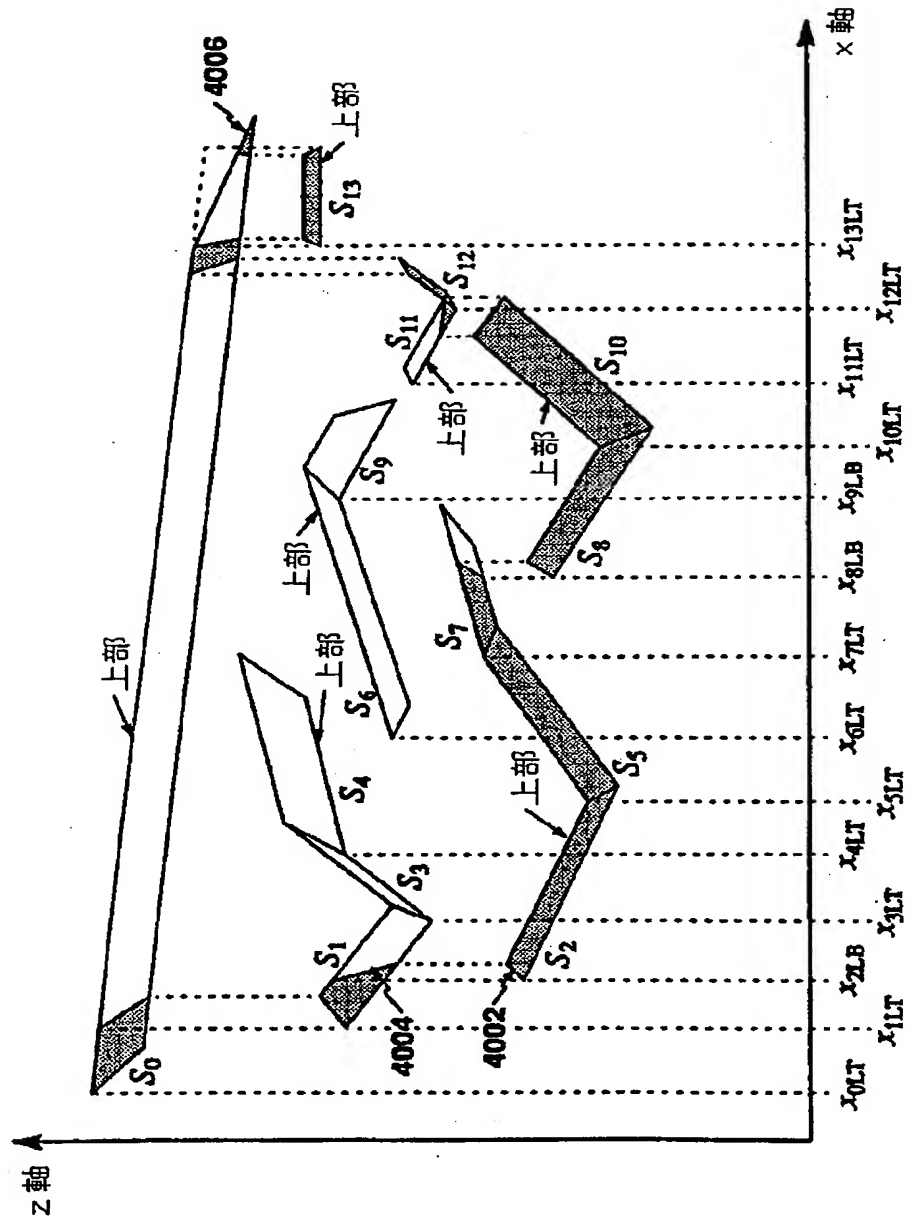
【図 39】

Figure 39



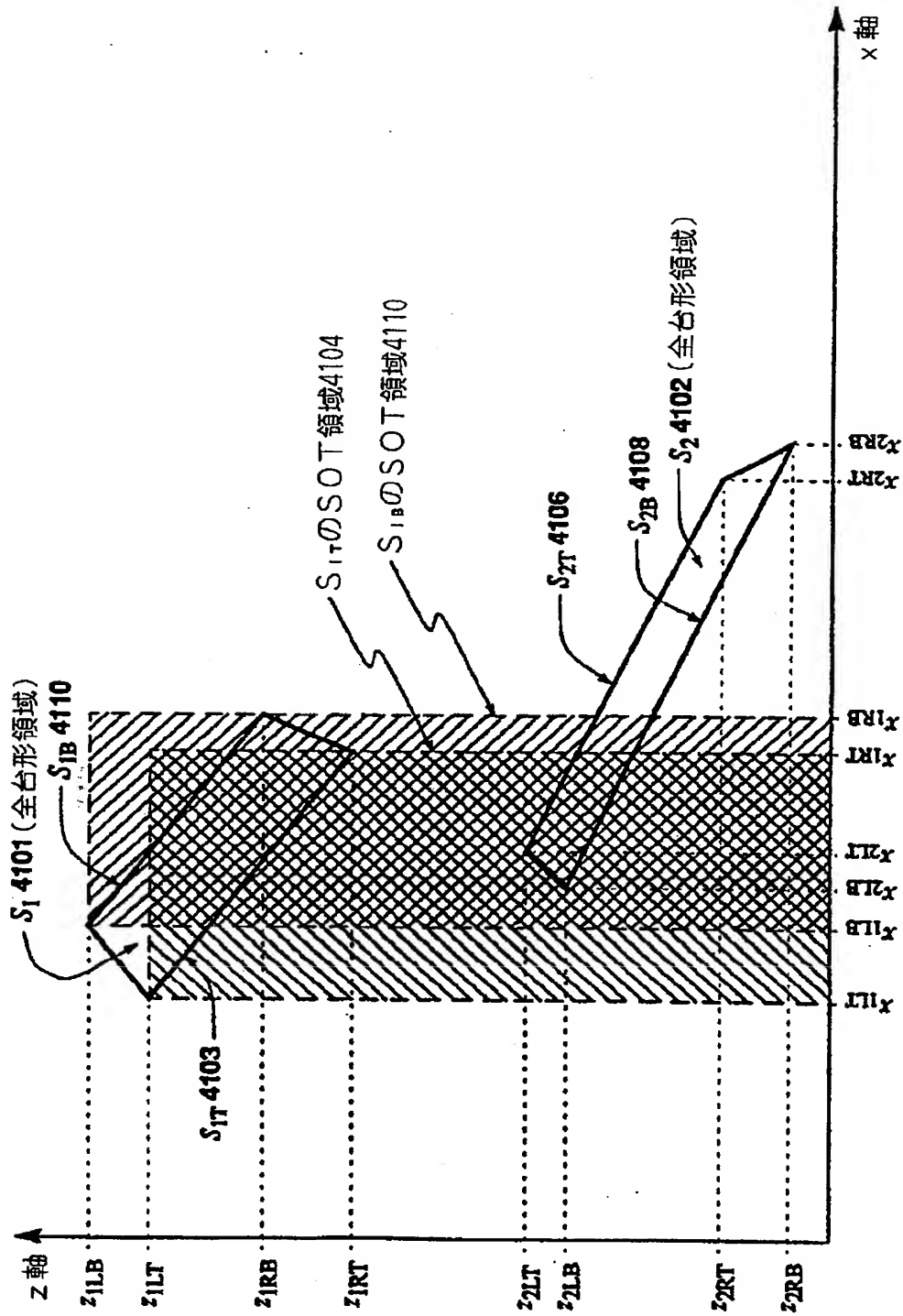
【図 40】

Figure 40



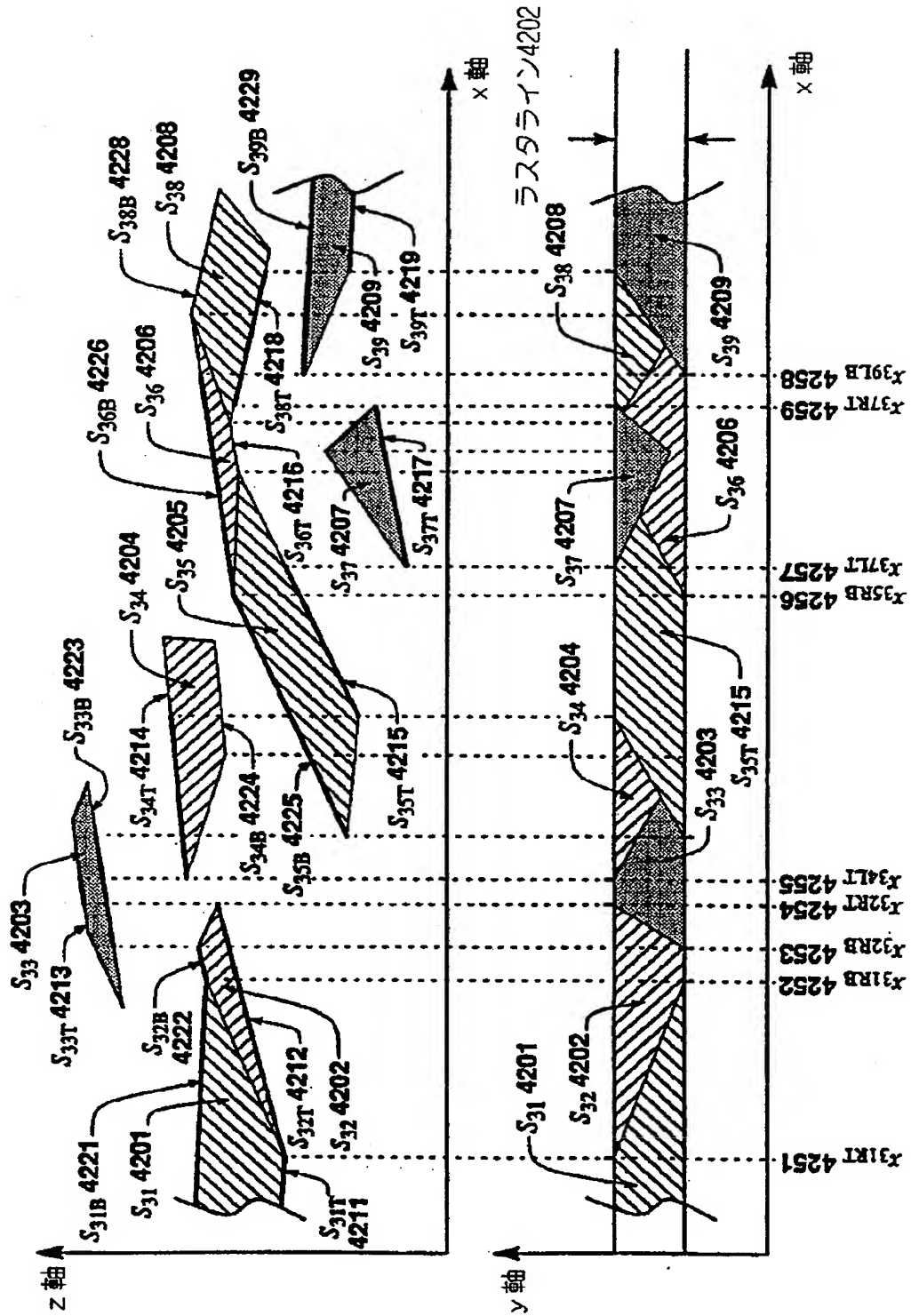
【図41】

Figure 41

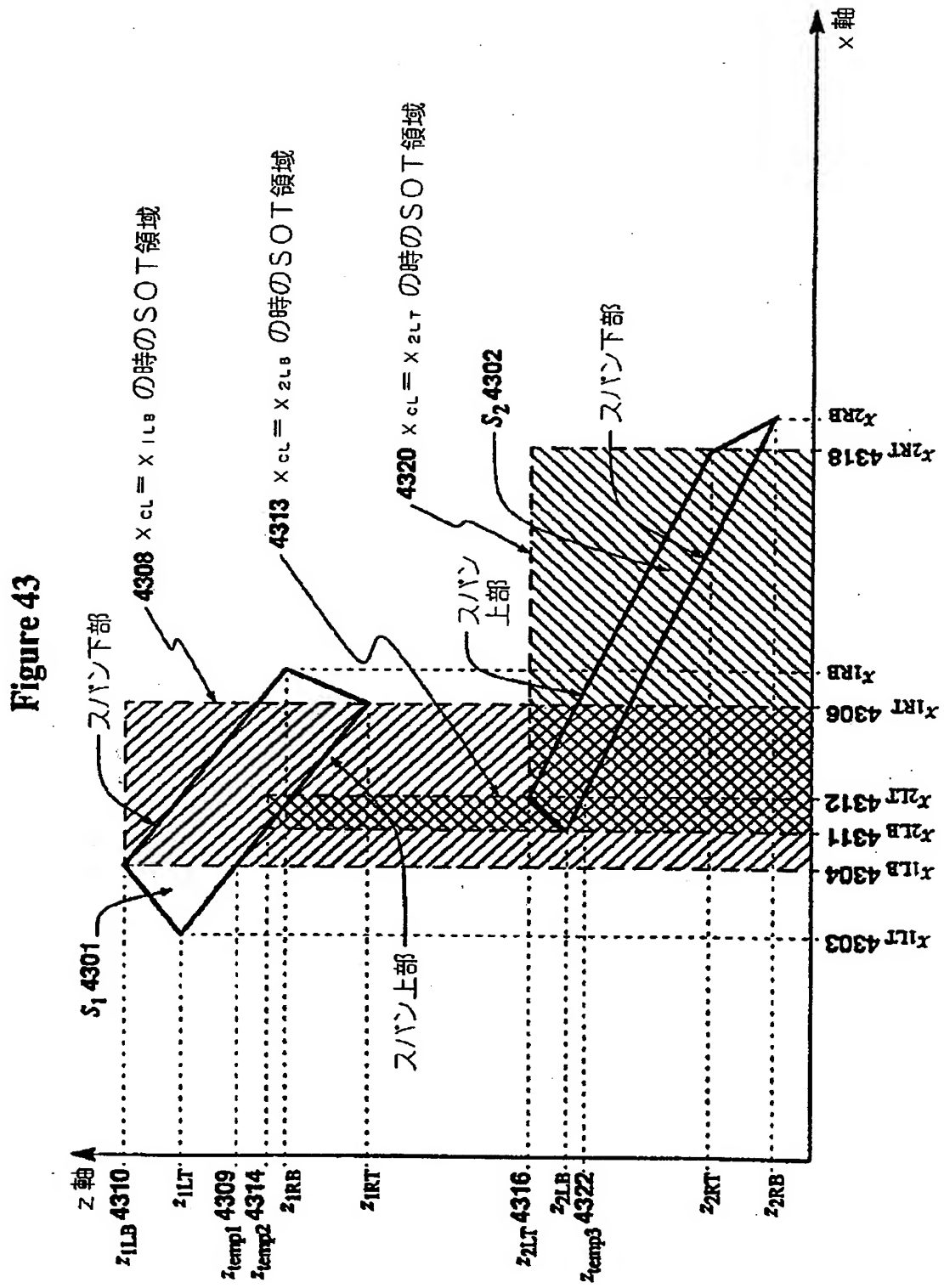


【図 42】

Figure 42

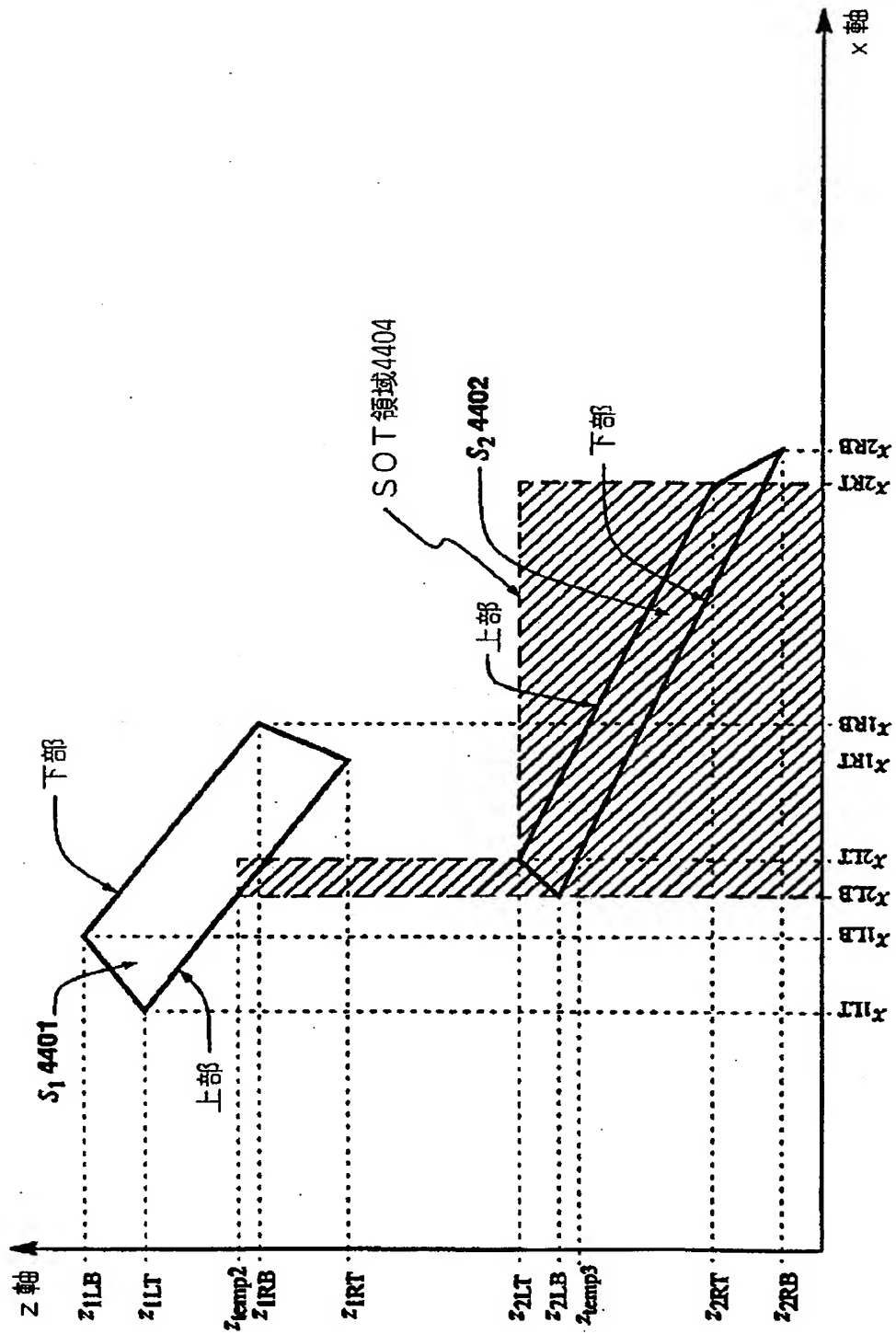


【図 4 3】



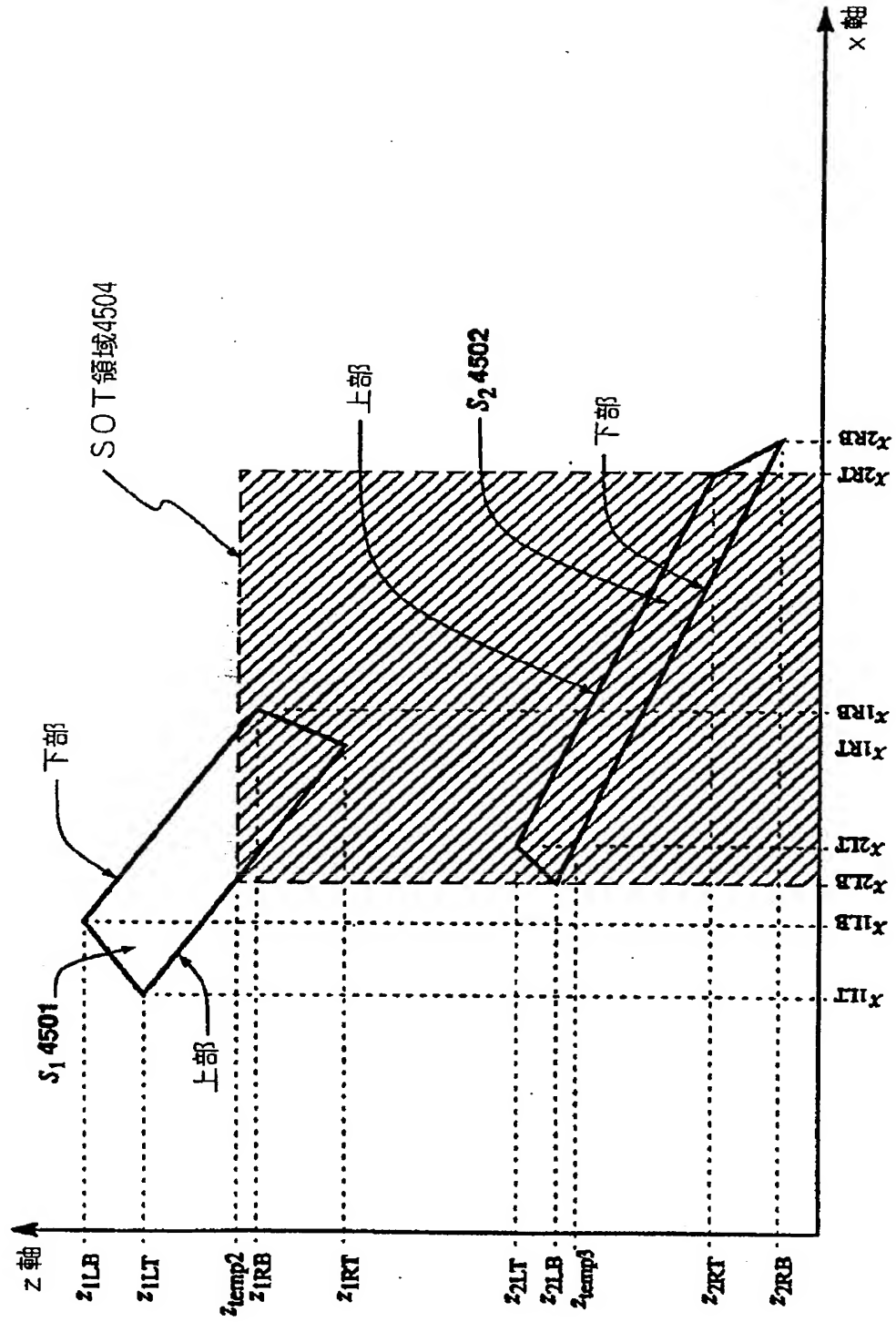
【図 4 4】

Figure 44



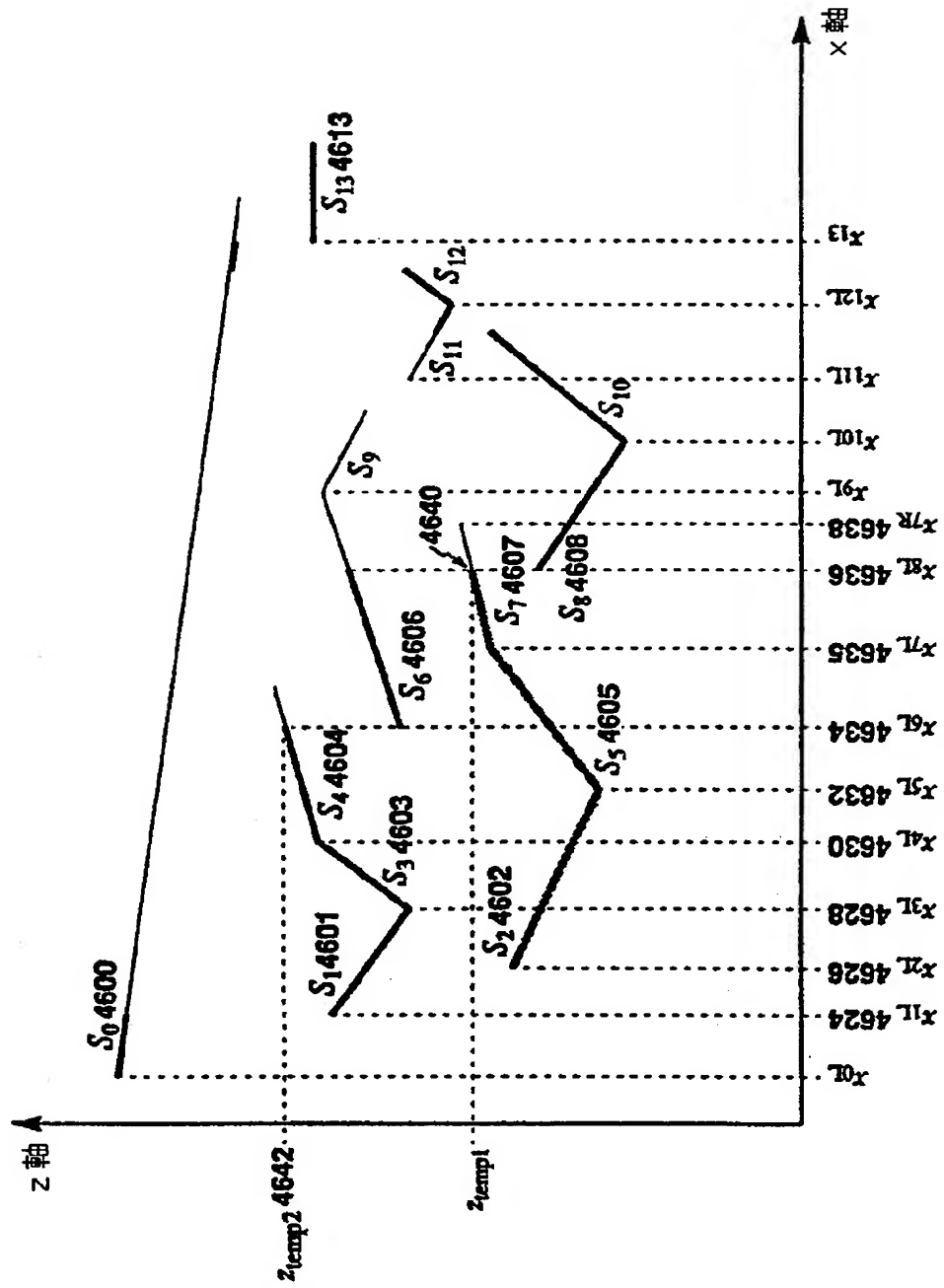
【図 4 5】

Figure 45



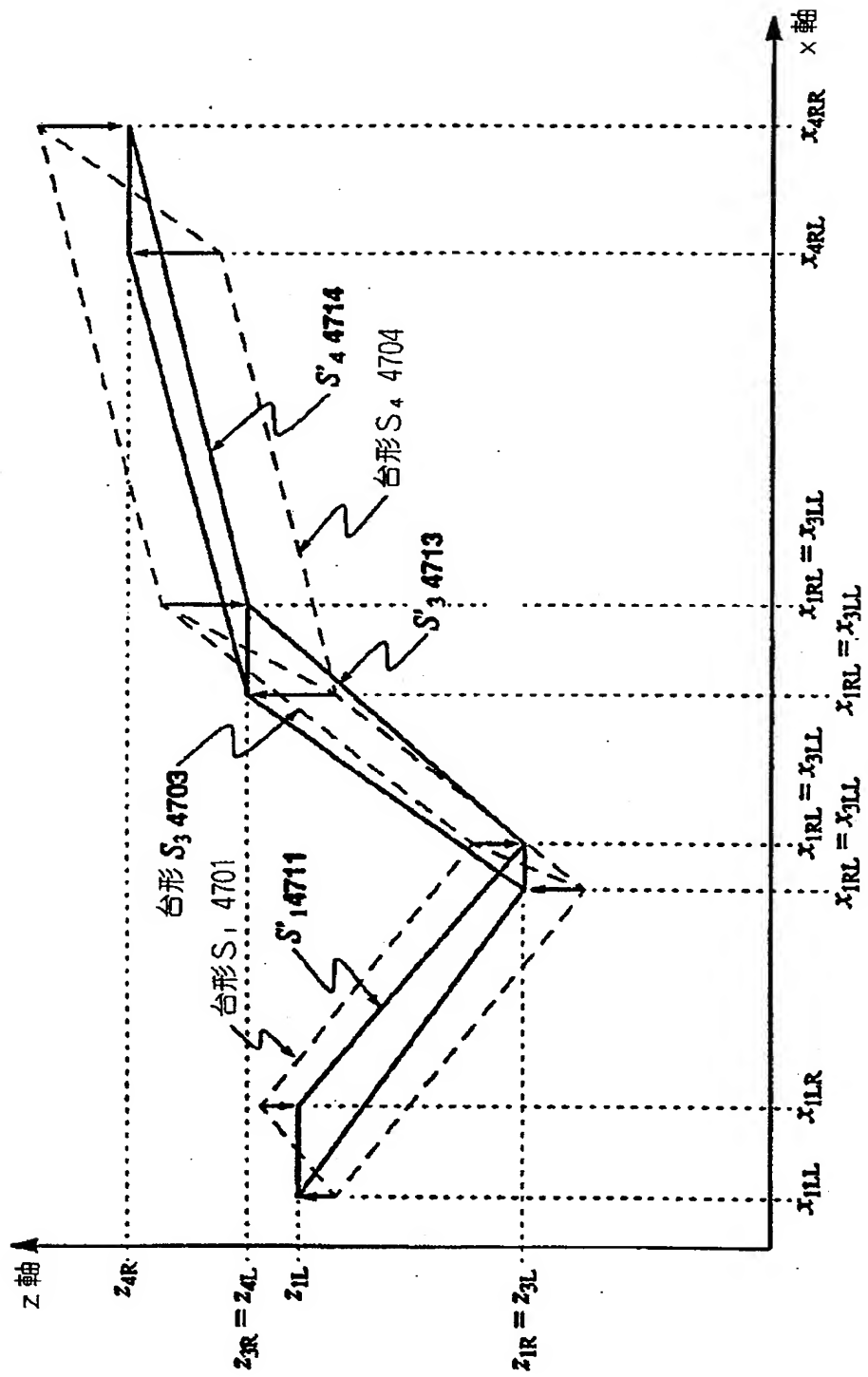
【図 46】

Figure 46



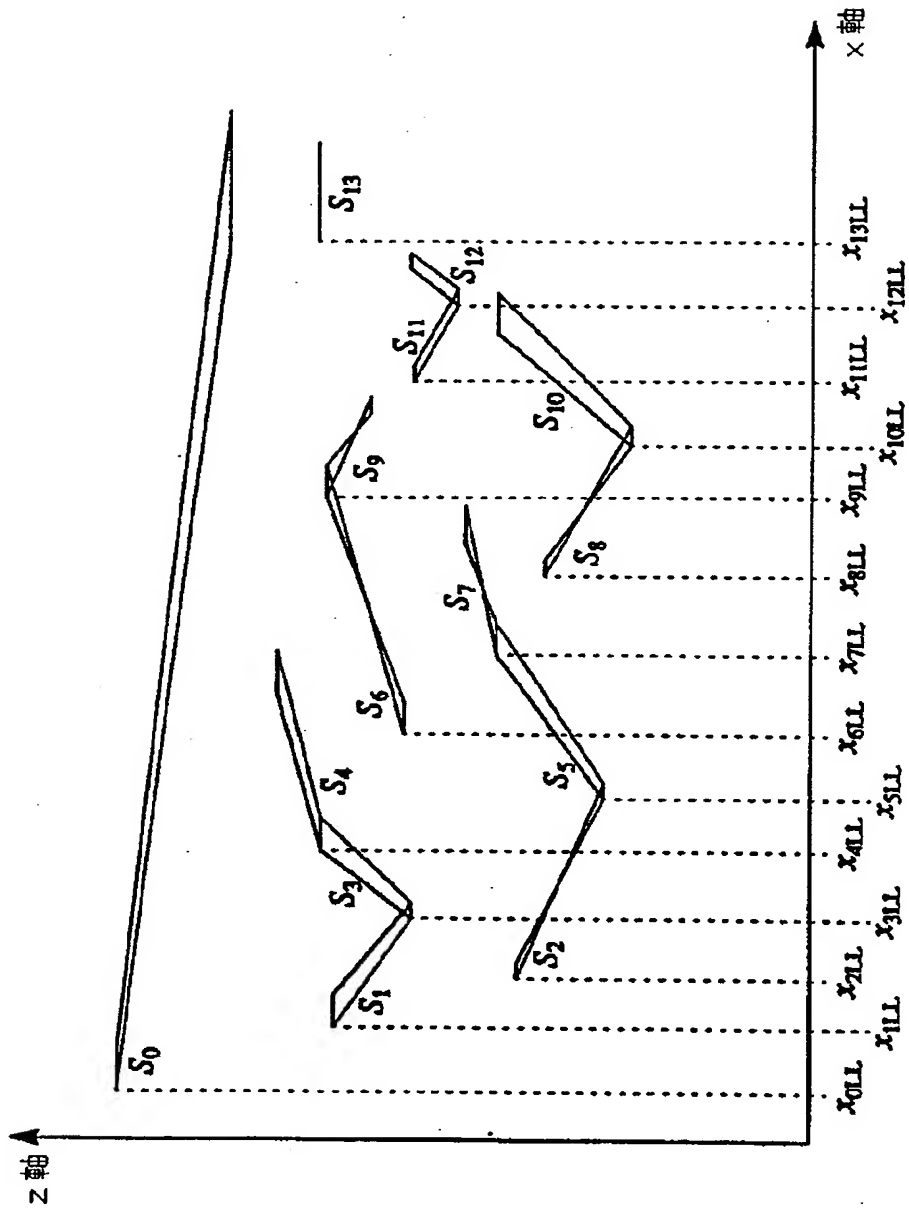
【図 4 7】

Figure 47



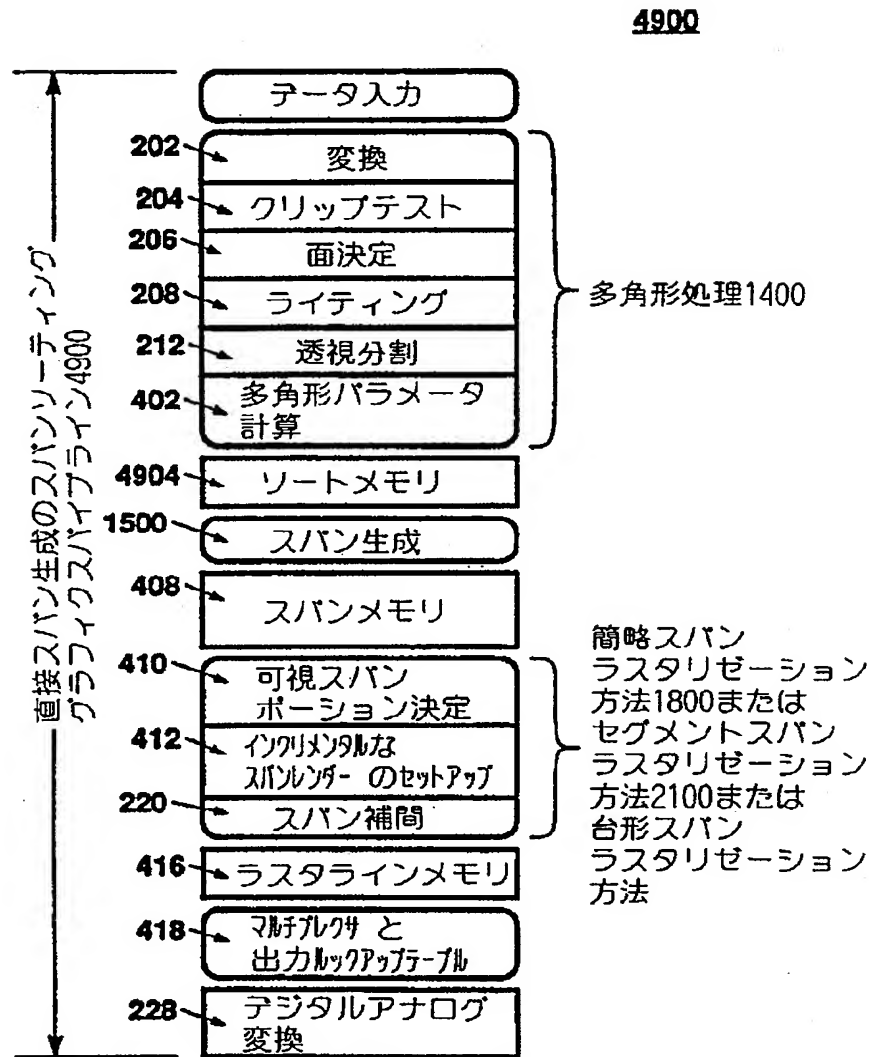
【图 4 8】

Figure 48



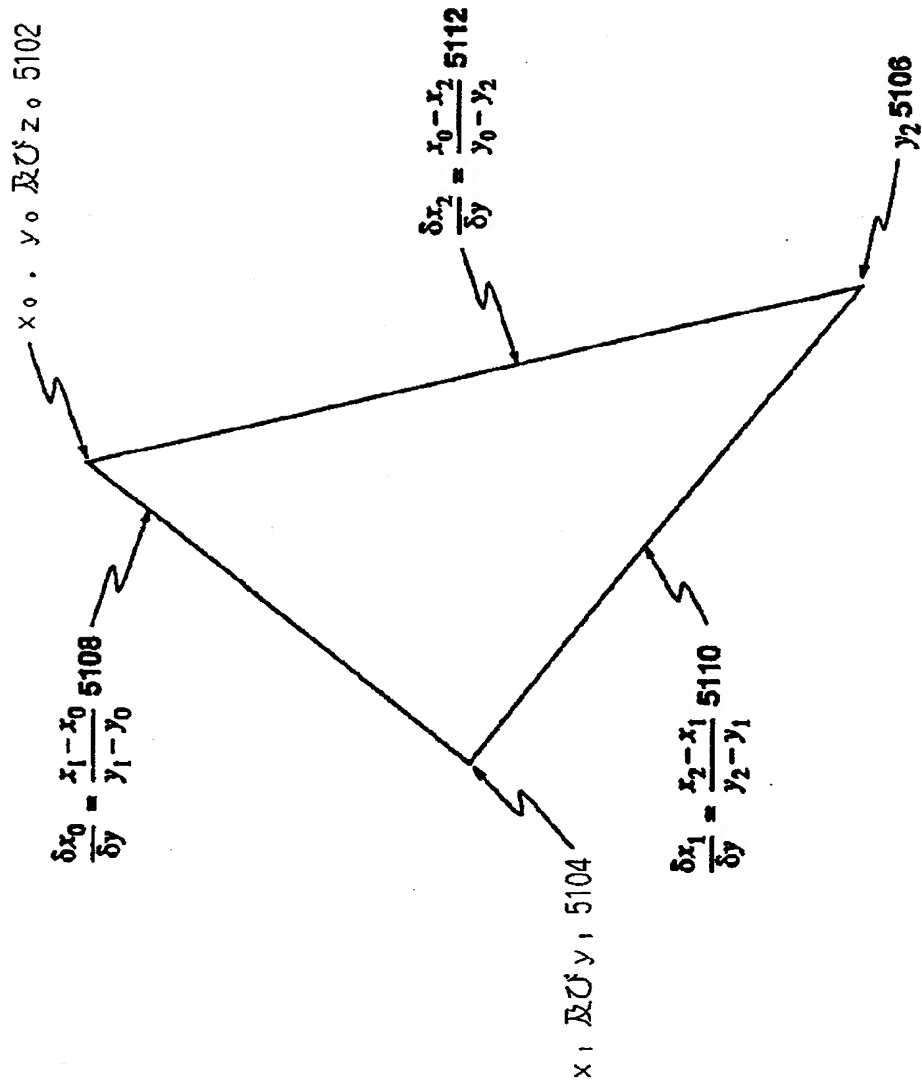
【図 4 9】

Figure 49



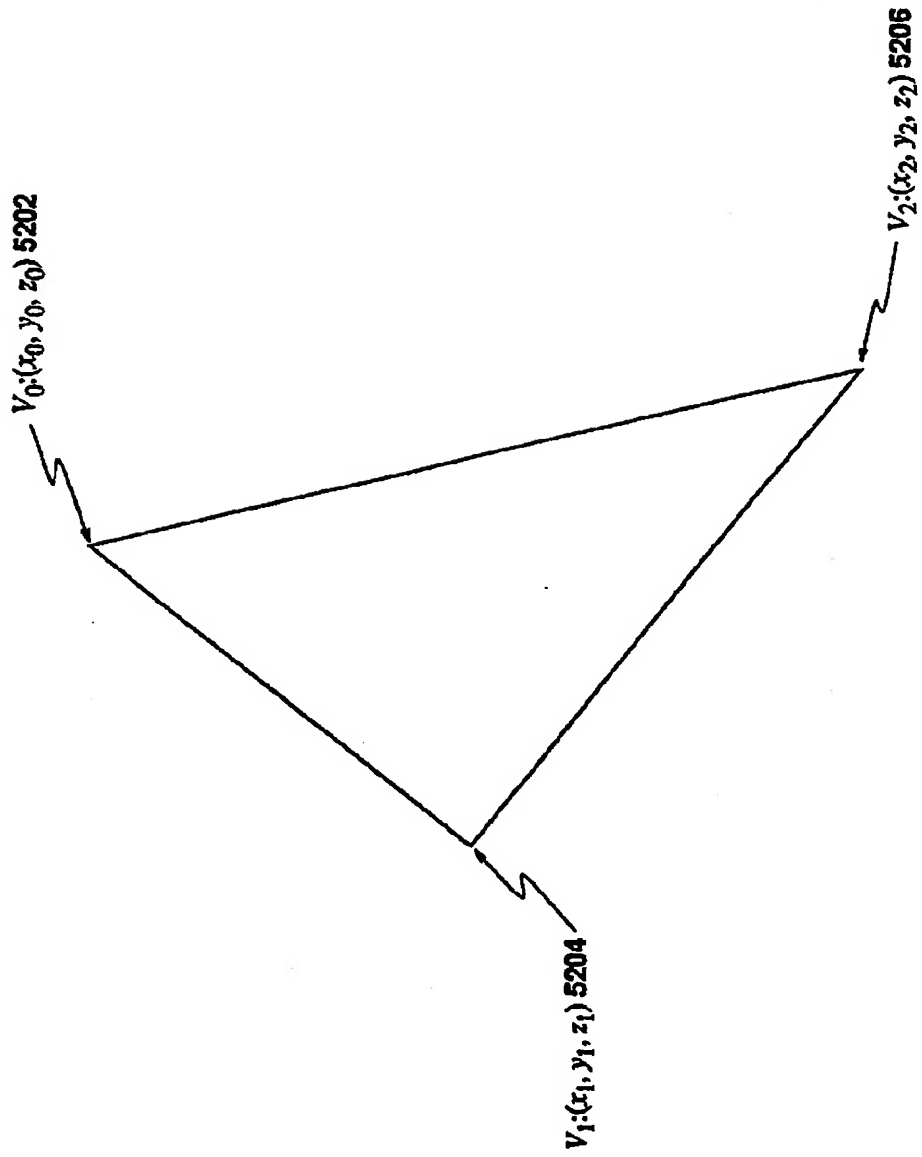
【图 5 1】

Figure 51



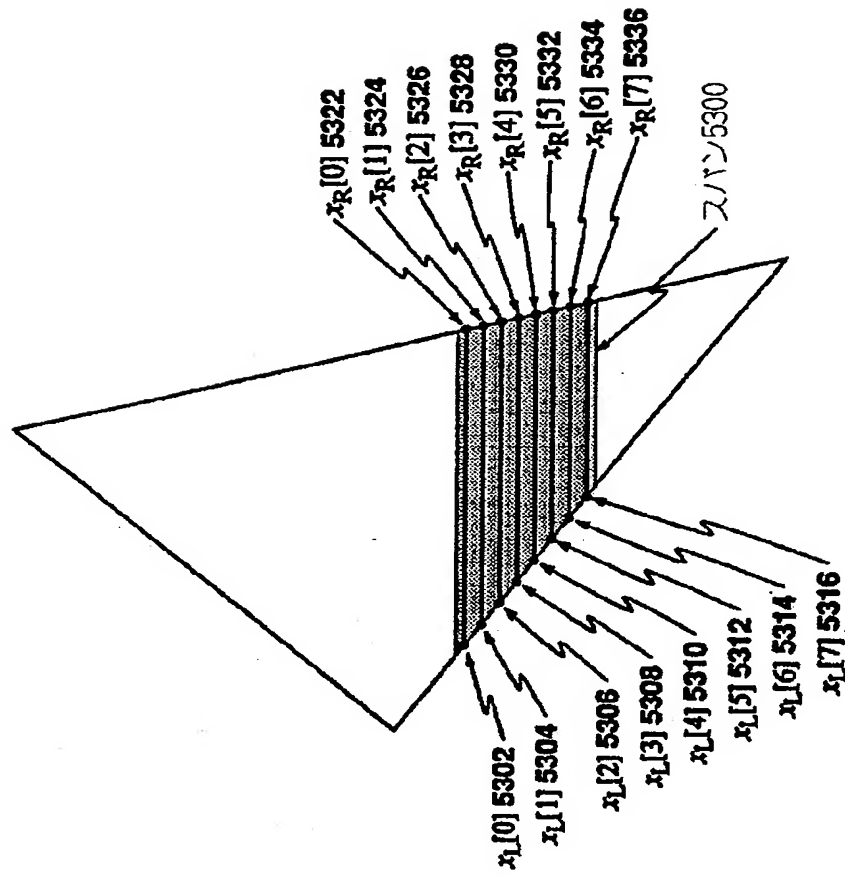
【图 5 2】

Figure 52



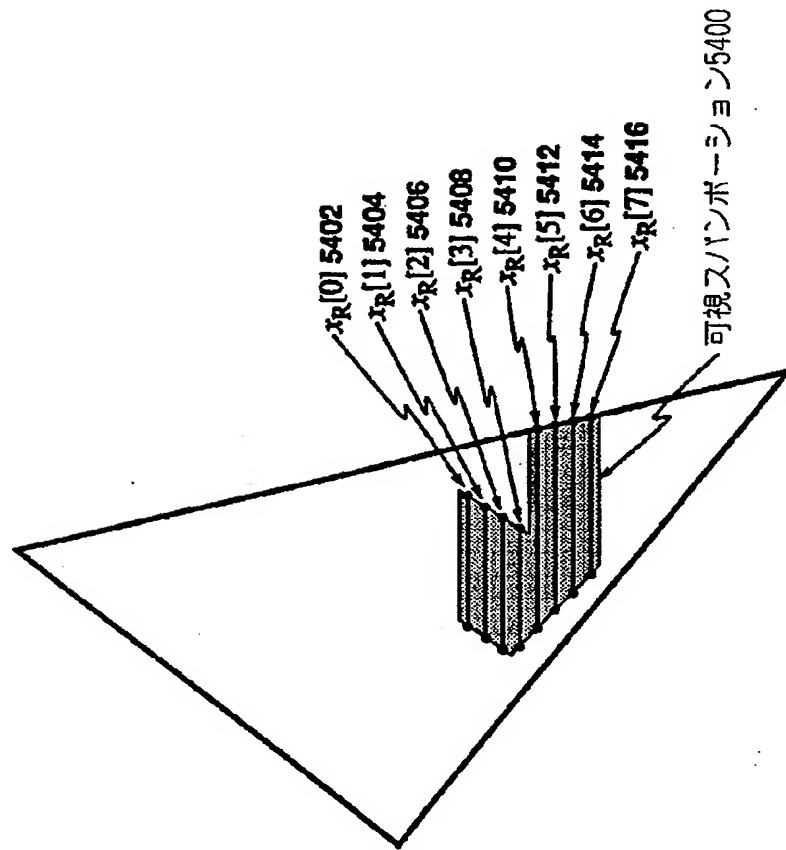
【図 5 3】

Figure 53



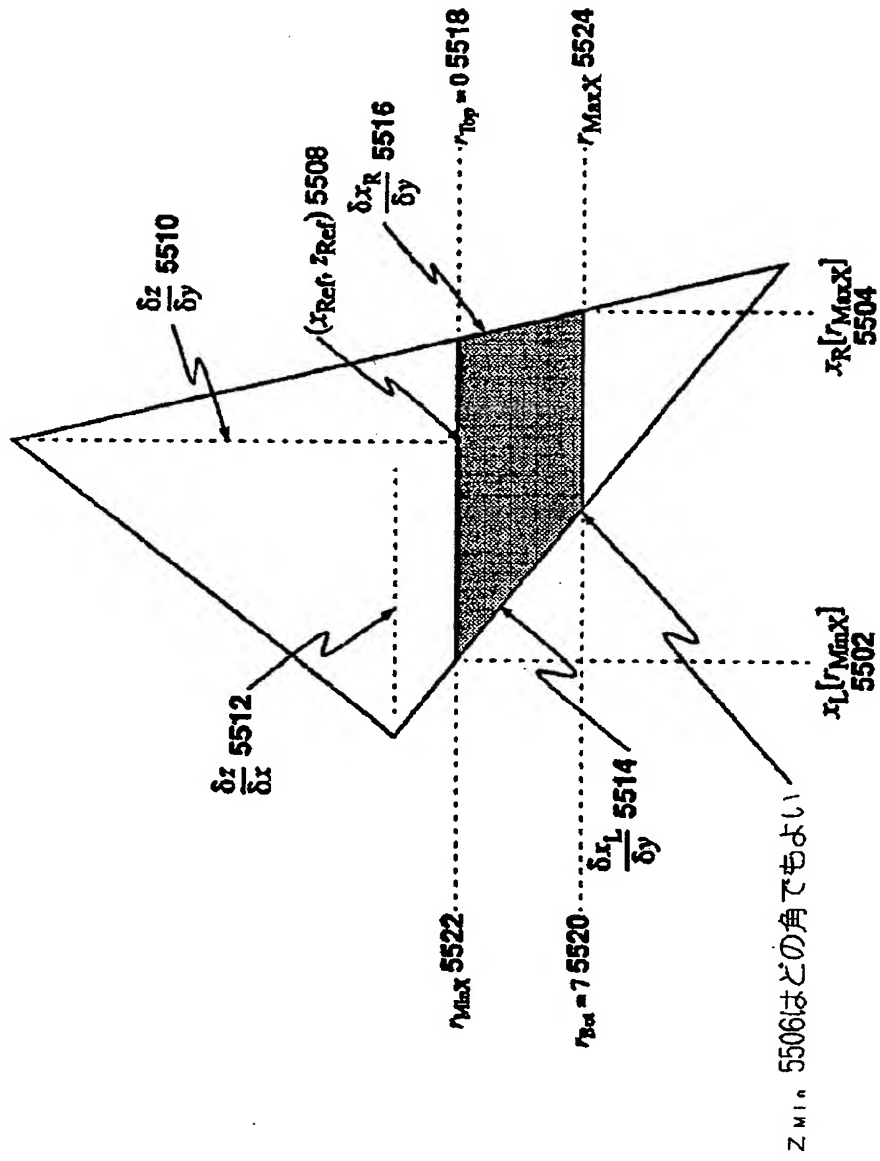
【図 5 4】

Figure 54



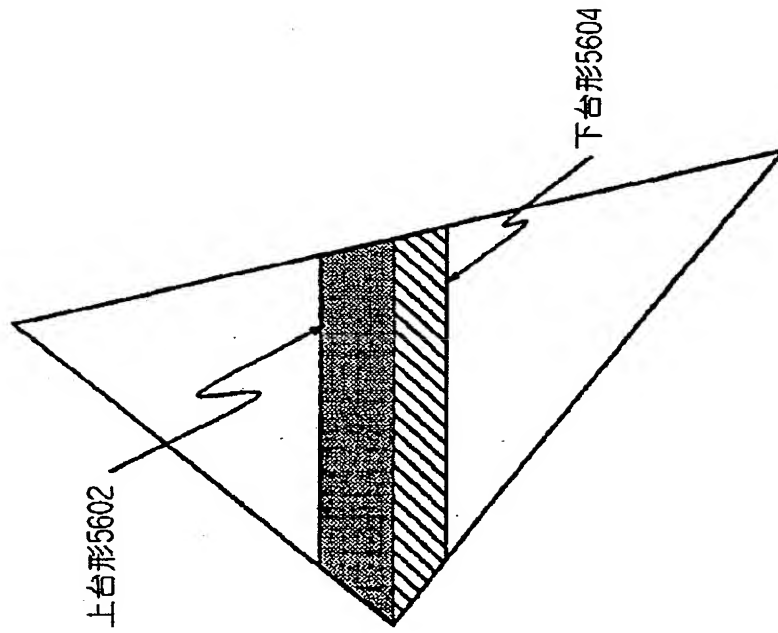
【図 55】

Figure 55



【图 5 6】

Figure 56



【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US96/12390

A. CLASSIFICATION OF SUBJECT MATTER IPC(6) : G06T 15/00 US CL : 395/119 According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) U.S. : 395/119 Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) APS, DIALOG, IEEE		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	FOLEY, et al, Computer Graphics: Principles and Practices, published in 1990 by Addison-Wesley Publishing Co., see pages 99, 108, 883-6, 903-4 and 980.	1-7
Y,P	US, A, 5,493,644 (Thayer, et al) 20 February 1996, see entire document, especially column 3, line 52 to column 4, line 53.	1-7
Y	US, A, 5,123,085 (Wells, et al) 16 June 1992, see entire document, especially column 5, lines 3 - 61, 52 to column 8, line 5 to column 10, line 64.	1-7
Y	US, A, 4,996,666 (Duluk, Jr.) 26 February 1991, see entire document.	1-7
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be part of particular relevance "E" earlier document published on or after the international filing date "L" documents which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "Z" document member of the same patent family		
Date of the actual completion of the international search 26 SEPTEMBER 1996		Date of mailing of the international search report 07 NOV 1996
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230		Authorized officer BRIAN A. HARDEN PARALEGAL SPECIALIST GROUP 2400 Telephone No. (703) 305-3800

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US96/12390

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US, A, 5,278,948 (Luken, Jr.) 11 January 1994, see entire document, especially column 2, line 31 to column 3 line 19 and column 6, line 38 to column 15, line 40.	1-7
Y	US, A, 5,129,060 (Pfeiffer, et al) 07 July 1992, see entire document, especially column 2, line 38 to column 4 line 9 and column 7, line 10 to column 68, line 39.	1-7
Y	US, A, 5,146,592 (Pfeiffer, et al) 08 September 1992, see entire document, especially column 2, line 40 to column 4 line 10 and column 7, line 14 to column 68, line 45.	1-7
A	US, A 4,825,391 (Merz) 25 April 1989, see entire document.	1-7
Y	US, A 5,394,516 (Wisner) 28 February 1995, see entire document, especially column 2, line 39 to column 3, line 32.	1-7

フロントページの続き

(81) 指定国 EP(AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OA(BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG), AP(KE, LS, MW, SD, SZ, UG), UA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IL, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, US, UZ, VN